DOCUMENT RESUME

ED 129 074 FL 007 984

AUTHOR	Montgomery, Christine A.
TITLE	Linguistics and Information Science. LINCS Project Document Series.
INSTITUTION	Center for Applied Linguistics, Washington, D.C. Language Information Network and Clearinghouse System.
SPONS AGENCY	National Science Foundation, Washington, D.C.
REPORT NO	LINCS-2-71P
PUB DATE	Jul 71
GRANT	NSF-GN-771
NOTE	108p.
EDRS PRICE	MF-$0.83 HC-$6.01 Plus Postage.
DESCRIPTORS	*Computational Linguistics; Information Retrieval; *Information Science; Information Storage; *Information Systems; *Language; Language Universals; *Linguistics; Linguistic Theory; Morphology (Languages); Semantics; Syntax
IDENTIFIERS	*Natural Language

ABSTRACT

The relationship between the disciplines of linguistics and information science has not yet been studied in depth. We must assess the state of our knowledge of natural language and determine how this knowledge is applicable within the context of an information system. The concept of a natural language information system can be specified in terms of the four components of acquisition, content analysis and representation, data management and information utilization. Morphology provides information science with its safest entree into the exploration of natural language. Systems have also been set up for syntactic and semantic analysis and for combining the two. The most solid achievements in computational linguistics involve syntax. The construction of a natural language information system clearly is not a trivial undertaking, for we are attempting to build a device for "understanding" natural language text before we fully understand natural language. It must be a joint attack made by linguists and information scientists. The common interest of both linguists and automated language processing specialists in natural language could offset their divergent analytical approaches and make them aware of the necessity of mutual cooperation in language processing projects. It appears that information science has gone about as far as it can go without linguistics, and vice versa. (CFM)

****
* Documents acquired by ERIC include many informal unpublished *
* materials not available from other sources. ERIC makes every effort *
* to obtain the best copy available. Nevertheless, items of marginal *
* reproducibility are often encountered and this affects the quality *
* of the microfiche and hardcopy reproductions ERIC makes available *
* via the ERIC Document Reproduction Service (EDRS). EDRS is not *
* responsible for the quality of the original document. Reproductions *
* supplied by EDRS are the best that can be made from the original. *
****

ED129074

FL007984

CENTER FOR APPLIED LINGUISTICS

LANGUAGE INFORMATION NETWORK AND CLEARINGHOUSE SYSTEM (LINCS)

LINGUISTICS AND INFORMATION SCIENCE

By Christine A. Montgomery

U S DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

All Rights Reserved

LINCS PROJECT DOCUMENT SERIES / NATIONAL SCIENCE FOUNDATION GRANT

LINCS #2-71 July 1971 NSF GN-771

CENTER FOR APPLIED LINGUISTICS, 1717 MASSACHUSETTS AVENUE, N.W., WASHINGTON, D.C. 20036

LINGUISTICS AND INFORMATION SCIENCE

by

Christine A. Montgomery

ACKNOWLEDGEMENTS

The support of the Center for Applied Linguistics, which provided in part for the preparation of this paper, is gratefully acknowledged.

Thanks are also due to my colleagues, J. L. Kuhns and R. M. Worthy, for the clarification of many of the notions presented here.

LINGUISTICS AND INFORMATION SCIENCE

In theory, the relationship between linguistics and information science is clear and indisputable: information science is concerned with all aspects of the communication of information, language is the primary medium for the communication of information, and linguistics is the study of language as a system for communicating information. In practice, however, the relationship between the disciplines of linguistics and information science has not been exploited.

It seems that there are two basic causes for this lack of interpenetration. In the first place, linguistics has had very little to offer in the area of semantics, or explication of meaning in natural language, and it is this aspect of language which is of most concern to information scientists.[1] Secondly, an explicit knowledge of how human beings receive and transmit information was practically unnecessary so long as information processing operations such as indexing were performed by humans. However, the introduction of automated information processing presents an entirely new set of requirements; no process which cannot be described in explicit detail is susceptible of automation in any meaningful sense.

Thus, in order to replicate the intellectual operations of an indexer, we must know precisely what these are. Since science has not provided us with a means of observing the neural activity of humans, we must select the alternative approach of simulating the intellectual operations of the indexer, based on the observable results of these operations and our knowledge about language as a system for communicating information. It is no easy task that confronts us, for we are attempting to use a machine to perform the activity of an intelligent human -- that is, to "understand" text: an operation which comes within the purview of the complex science of artificial intelligence.

There are two main incentives for undertaking this formidable task. The first of these is the ever-growing volume of information which seems an inevitable adjunct of our complex civilization; the second is a consequence of the first and of human frailty. That is, because of the increasing flow of information, the harassed human indexing analyst is under pressures that hinder the effective performance of his task. Thus, in many cases, equaling the performance of a human with a computer becomes considerably less formidable than it might seem.[2] In the interest of advancing information science as well as linguistics, however, it is the more difficult task of equalling an ideal standard of human

performance in processing natural language information which we must undertake.

In order to estimate the magnitude of the task, we must assess the state of our knowledge about language and determine how this knowledge is applicable within the context of an information system. To this end, I shall first outline a generalized concept of an information system, indicating where efforts in information science have concentrated, what has been lacking in these efforts, and how linguistic knowledge can be utilized. This discussion will be followed by an assessment of the current state of knowledge about language as reflected in computational linguistic models and techniques, and in linguistic theory. Computational linguistics, which is treated in the form of a lengthy state of the art survey, is here given priority, assuming that computable concepts have the most immediate relevance to the design of an automated system for "understanding" natural language text. Finally, based on the state of our knowledge about information in natural language form and the level of development in the science of information systems, conclusions as to the role of linguistics in information science are presented and suggestions for cooperative efforts are outlined.

THE CONCEPT OF A NATURAL LANGUAGE INFORMATION SYSTEM

The concept of a natural language information system can be specified very simply in terms of the four components

of acquisition, content analysis and representation, data management, and information utilization, as presented in Figure 1. The acquisition component includes the selection of an appropriate subset of the universe of information and the introduction of this subset into the particular system. Information records acquired by the system may be documents or document surrogates or subsets, facts or data items--all are "packages" of natural language information differing in size, and, in some respects, in their internal construction.[3]

These information records must then undergo a process by which their content is analyzed and represented in some standard form, which is accepted for processing by the data management component. The user interacts with the system through the content analysis and representation component, which passes his requirements to a data management executive that provides responsive output. The utilization of this information by the requestor is represented by the fourth component, which may itself involve a complex subsystem for storing and processing data. This component impacts on the components of the main system through a feedback loop. Another feedback loop links information generated by the data management system to the components of acquisition and content analysis and representation.

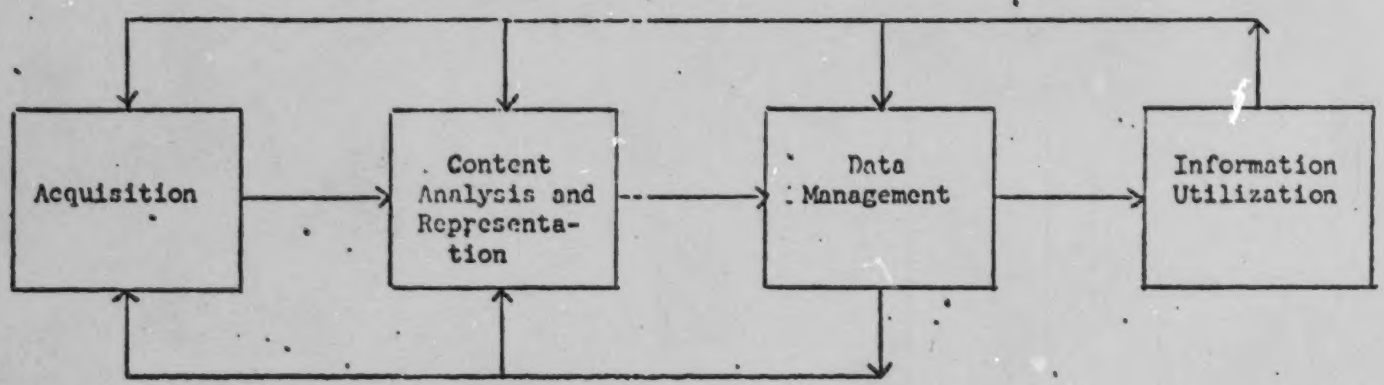

Figure 1. Components of a Total Information Systems Concept

A schematic of the content analysis and representation component is presented in Figure 2. The natural language information records acquired by the system as well as the natural language information requirements generated by the user are translated into some formal language which is thus the medium for communication between the user and the system, as well as the basis for communication within the system. The formal language might simply be the system of content categorization represented by a subject authority list, or in a more advanced application, a complex system which specifies the syntactic and semantic content--e.g. some enriched version of the propositional calculus.

In any case, the process of content analysis ideally involves the identification of the concepts contained in the information records and requirements, and the determination of the relations linking these concepts. The first procedure is based on some kind of semantic analysis, and the second on a syntactic analysis; the two types of analysis are highly interdependent, and in an automated content analysis system with such subcomponents, it is clear that the translation operation is not a trivial problem, in spite of its modest representation in the schematic. In fact -- assuming an automated system-- it is precisely at this point that linguistic competence is necessary, for the translation

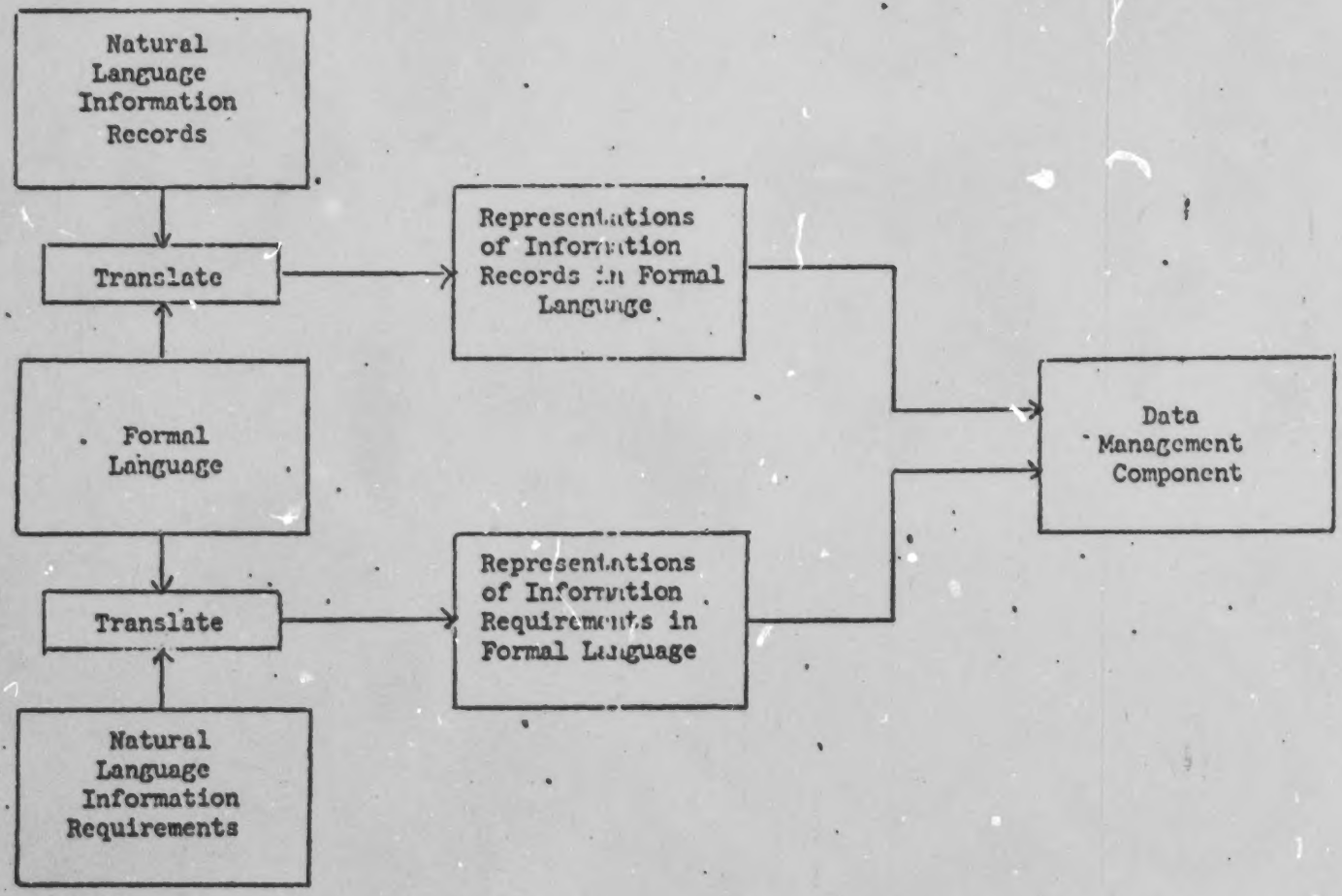

Figure 2. Component for Content Analysis and Representation

operation consists in understanding the content of the natural language records and requirements and specifying it in terms of the formalism which constitutes the internal language of the system.

Like the information records, the natural language information requirement of the user may also take various forms; the model in Figure 2 is purposely generalized in order to represent a variety of types of information systems, from document retrieval to question-answering. The various forms of the user information requirements include specific and general requests, requests made at a particular time, or requests of long duration -- say, user interest profiles (from this point of view, the retrieval and dissemination operations are analogous). Another possible form of user information requirement is a continuous interaction with the information system in an on-line mode. To summarize, the formalism of the content analysis and representation component specifies the means of communication between the user and the information store, whereas the subsystem for search and retrieval embodied in the data management component specifies the mode of communication between the user and the information store.

A generalized model of the data management component is presented in Figure 3. Within this component, the data

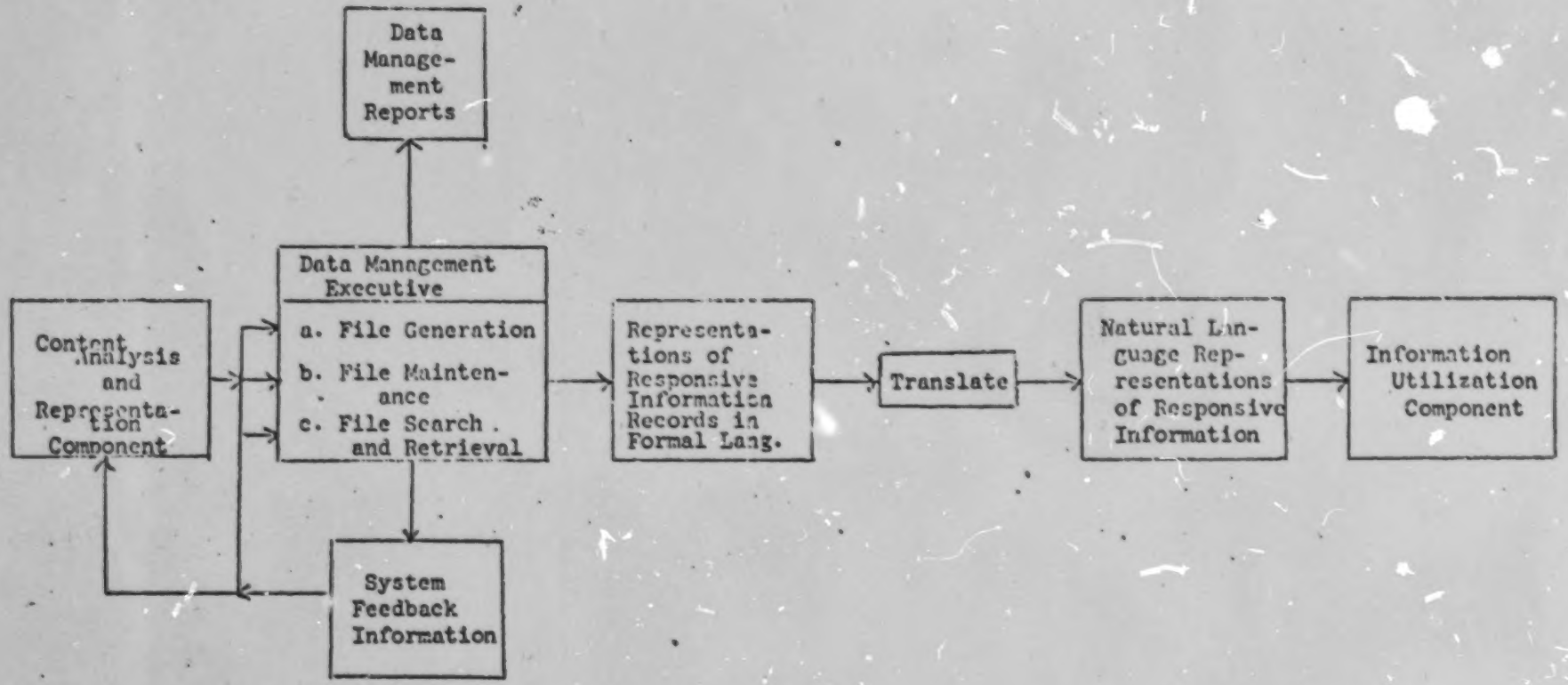

Figure 3. Data Management Component

management executive is the automated file clerk, accepting content representations of information records and requirements from the content analysis and representation component and interpreting these as storage and search commands respectively. In the case of a document (or abstract) retrieval system (as currently conceived), the storage operation would presumably involve the creation of an entry in an inverted file using the content representation of the natural language text, as well as the addition of the machine-readable text to some sequential file ordered by date of accession. Searches would be performed on the inverted file (unless accession numbers were specified) and documents retrieved by matching accession numbers generated by search of the inverted file against the sequential file. The "translation" between the representation of the document in the formal language of the system -- say, as a string of descriptors -- and its natural language representation would thus be effected by a simple matching procedure, inasmuch as separate files of both representations are maintained internally.

In the case of a fact retrieval or question-answering system, the storage procedures may manipulate more complex data types, and several different types of files may be required

for the data base. The search procedures are inevitably complex, and may involve elaborate strategies for extracting implicit information from the data base. Moreover, files of the natural language representations of the information records may not be maintained, in which case the "translation" from the formal language of the system involves the generation of natural language output. As in the case of "translation" from natural language to the formalism during the content analysis operation, the role of linguistics in an information system emerges quite clearly. What is less clear is the case for linguistic involvement in the design of the data management executive itself--specifically, in the design of file structures and search strategies.

In any event, knowledge of the language of the particular universe of discourse is critical to system design and operation; the data management executive must therefore provide surveillance of natural language information entering the system to insure continuous refinement of the system's information processing capability (as indicated by the feedback loop in Figure 3). At a very basic level, information developed by the data management executive for system improvement may consist simply of content word frequency statistics; these data are then studied by linguists and other analysts, and the results are incorporated in modifications to the content analysis operation. Ideally, such data would include various types of

frequencies based on syntactic constitutes, i.e., sentence and phrase types, as well as conceptual constitutes, for natural language records and requirements analyzable by the system, and detailed diagnostics for records and requirements unanalyzable by the system at the given stage of development. In addition, the data management executive generates other types of feedback information based on the representations of information records and requirements in the formal language, e.g. number of items in the various files, system usage of file items, number and type of searches carried out.

Using this generalized concept of natural language information system, it is possible to characterize attempts at automation of natural language information processing.

With respect to the component for content analysis and representation, approaches developed by information scientists have for the most part concentrated on statistical content analysis of documents and document collections. Some of the more recent efforts along these lines are described in the following section under "Automated Semantic Analysis."

On the whole, however, attention has tended to focus almost exclusively on data management operations of various types, including automation of the procedures for generating and maintaining files of subject headings, descriptors, or thesaurus entries. Most automated retrieval systems are constructed around some version of the data management component.

For example, consider the various formatted file systems, some of which handle extremely large files of data elements, but only in the fixed format specific to the system. In systems of this type, data acquisition may be semi-automatic, but content analysis is manual; information records are segmented into a set of data elements, only a small number of which may serve as content parameters for retrieval.[4]

Other approaches to automated retrieval feature unformatted text files. Although the emphasis is still on data management, the data in this case are strings of natural language text. This approach is exemplified by systems such as those of the Data Corporation and Aspen Systems Corporation. In these systems, acquisition is generally semi-automatic: for example, keyboard-to-tape devices may be used to code input text, which is then sorted and stored as an inverted file containing all content words. Whether the system is on-line or batch, the operation of content analysis devolves upon the user, who must--in the case of the Aspen system--fill out an elaborate search request involving the specification of the required information in several forms, the identification of synonymous terms, term frequencies, word stems, grammatical categories, and other parameters.

One of the most effective systems of this type is SPIRAL, an automated text retrieval system developed at Sandia Laboratories (West 1968). SPIRAL retrieves relevant documents or document paragraphs based on key word, phrase, or paragraph request formats. The system also contains a unique "re-inquiry" option, which allows recycling of paragraphs obtained through key word or phrase requests to retrieve other paragraphs containing the same vocabulary items. This provides an automatic technique for expanding terms of the original query, and hence relieves users of some of the burden of content analysis.

However, one of the objections which can be raised against systems which charge the user with the intricate task of generating all the content representations for the search (e.g. the Aspen system), as well as systems which attempt to lighten the user's burden by some automatic means (e.g. SPIRAL) concerns their long range operating efficiency. Since the data base is effectively re-indexed for each request -- regardless of whether the request is exactly the same as another submitted previously -- systems of this type would appear inefficient in the long run.

To summarize, we find that in terms of the information systems concept presented in Figures 1, 2, and 3, information science developments have largely concentrated on automating

operations of the data management component. In contrast, efforts to automate operations of the component for content analysis and representation have been relatively few and far between. In view of the fact that it is precisely this component which requires understanding and specifying the content of natural language text, it is hardly surprising that information scientists have been somewhat reluctant to undertake such a difficult task.[5]

As was noted above, content analysis involves syntactic and semantic analysis of natural language, and some experimental approaches at automating these operations have been developed under the label of computational linguistics -- an interdiscipline concerned with the automation of language processing operations. A critical survey of these approaches is presented in the following section.

COMPUTATIONAL LINGUISTICS: A SURVEY OF MODELS AND TECHNIQUES

Automated Morphological Analysis

Morphology, or the study of word formation, provides information science with the safest entree into the exploration of natural language during the present period of turbulence in theoretical linguistics. Morphology has been the least debated aspect of the theory of grammar since Chomsky became concerned with theory in linguistics,[6] and in contrast with

syntax and semantics, it is clearly the least debatable.

Automated morphological analysis can be an extremely useful tool in information processing, and it is rather surprising that it is rarely exploited. Generally speaking, a morphological analyzer for an automated system consists of an affix stripping algorithm together with some few tables of words constituting exceptions to the rules used by the algorithm. (For example, in English, the -ed in "seed" must be distinguishable from the -ed in "heated".) There are four major uses for a morphological analysis algorithm in natural language data processing, three of which are directly relevant to any information system which processes natural language text and maintains word lists of any kind.

First, as was noted above in developing the concept of an information system, there is a need for continual analysis of the particular subset of a language which constitutes the universe of discourse for the given information system. At the very least, statistics on occurrence of individual words are necessary to identify high frequency words which are peculiar to the given universe of discourse and thus require special treatment. If no morphological analyzer is used, a "word" is simply a unique character string, and occurrences of, say, "computer" and "computers" are calculated separately. Such a procedure is certainly counter-intuitive, and the output is

inevitably of rather dubious utility. On the other hand, if a morphological analysis algorithm is utilized together with an arbitrary three-word definition of a phrase, it would be possible to derive a statistical phrase group like the following, which would indicate a conceptual relation of potential interest:

theoretical linguistics, linguistic theory,
theorists in linguistics, linguistic theorizing.

A second major area where a morphological analyzer is useful is in the compression of voluminous word lists or dictionaries maintained in a given system. In dealing with highly inflected languages such as Russian or Finnish, compression is essential, as a full form dictionary can be 10 or 20 times the size of a stem dictionary.

A third use of morphological analysis is the converse of the second -- that is, the automatic expansion of terms in a query or search prescription to the full paradigm.

A fourth use of an automated morphological analyzer is in the identification of grammatical categories in text processing systems employing some form of syntactic analysis. In attempting to process a text sample of any size, many of the words are inevitably lacking in the dictionary. These must be assigned grammar codes to insure that the syntactic analyzer does not

grind to an untimely halt for lack of appropriate grammatical information. The limited syntactic analysis systems of Baxendale and Briner (see below) utilize an algorithm of this type. Fairly elaborate analyzers have been developed for English by Earl (1967) and Chapin (1968). The former claims a 95 per cent successful assignment of grammatical categories based on suffix; however, the assignments are not necessarily unique.

In addition, many aids to morphological analysis -- some of them involving large corpora in machine-readable form -- are currently available. For English, there is Dolby & Resnikoff's "Word Speculum" (1967) which comprises five volumes, including forward and reverse word lists (volumes 2 and 3), and a reverse part of speech word list (volume 5.) There is also a reverse dictionary of French, compiled by Juillard (1965). Papp (1967) reports a study which involved the transfer of 60,000 entries of the "Dictionary of Definitions of the Hungarian Language" onto punched cards. A morphological dictionary of Russian has recently been compiled by Worth, Kozak, and Johnson (1970). Schnelle's group at Bonn has adopted a somewhat different approach, creating instead a machine-readable list of approximately 3,000 basic morphemes in German and writing rules for generating derived forms (Bünting 1969).

Automated Syntactic Analysis

Systems for Limited Analysis

Before discussing systems for full-scale syntactic analysis, it is interesting to review for comparative purposes a few systems which aim at identifying only those sentence elements considered most relevant for content analysis. This line of thought has been followed since the late 1950s by Phyllis Baxendale, who began with an attempt to replicate the operations performed by persons in skimming material to identify significant sentences and sentence elements -- specifically, noun phrases. In a more recent paper (1966), Baxendale and Clarke describe the details of a computer program for a limited syntactic analysis of English text. Grammar codes are assigned by matching input items against a lexicon which contains common function words, a suffix dictionary, and exceptions to the suffix stripping rules. Items which cannot be resolved by these rules are given the arbitrary classification noun/verb. Following dictionary lookup, the program first uses phrase-bracketing rules to identify phrases which can be nominal, verbal, prepositional, participial, gerundive, adverbial, and adjectival. The program next uses relative pronouns and subordinating conjunctions to identify clause beginnings, and finally applies a set of rules called "sentence-hood testing rules" to verify whether the correct interpretation of items

labeled noun/verb has been made (these are mainly tests of number agreement between potential subjects and verbs).

Kravchenko (1968) describes a less elaborate system design for identifying nominal and adjectival phrases as semantic elements for development of dictionaries characterizing particular universes of discourse, as well as for use in automated abstracting. A manual simulation of an automated indexing experiment uses this algorithm (augmented by a morphological analyzer) as a test of its effectiveness in identifying occurrences of nouns which may serve as role indicators for descriptors (Otradkinsiy & Kravchenko 1969).

Klingbiel (1969) reports an automated experiment with a similar goal -- namely, the use of a limited syntactic analysis to identify index terms. Assignment of unique syntactic codes is based on function of the item within the system, as specified in a "disposition" dictionary where each entry is an element pair consisting of a lexical item and an address of a macro instruction which supplies the appropriate code for the given item (e.g. noun, adjective) and performs other operations as required. Syntactic codes identifying potential index terms are accumulated in a register and the resulting string of codes is matched against a list of syntactic formulas specifying acceptable configurations for index terms.

An approach similar to that of Baxendale (and presumably related) is developed in Briner's SYNTRAN (1968), which is, however, a program constructed for operational rather than experimental use, and can thus be expected to exhibit more efficient operating characteristics. The basic premise of SYNTRAN assumes that nouns which are characterized by a variety of syntactic functions (specifically, those which function as subject, object, and modifier) are the most significant in a text, provided their frequency of occurrence is also statistically significant. The limited syntactic analysis is accomplished by a three-step procedure. First, word endings and common words are looked up to assign appropriate grammar codes. Sentences are then segmented into two types of word groups: a) those initiated by prepositions, articles, adjectives or nouns; b) those initiated by verbs or adverbs. Based on these word groups, presumable syntactic functions are determined for nouns using the following criteria: a) a noun which precedes a verb is a subject; b) a noun which follows a verb or preposition is an adjective.

Obviously, syntactic decisions based on such criteria can only be correct part of the time. However, the percentage of correct decisions may be higher than the theoretical linguist would like to admit. An article by Clarke and Wall (1965)

compares the performance of the Baxendale-Clarke-Wall system and that of a modified 1963 SHARE version of the Kuno-Oettinger Syntactic Analyzer (see the following section) in identifying well-formed phrases of a test group of sentences, noting that the average percent of success was 92 for the limited parser as against 85 for the Syntactic Analyzer.

These statistics do not necessarily mean that the Baxendale-Clarke-Wall system was a complete success; rather, they reflect on the inadequacy of the Kuno-Oettinger Syntactic Analyzer. Clearly a syntactic analyzer which is limited in scope -- as are all those mentioned above -- is also limited in effectiveness. Although they may function adequately in terms of restricted objectives, some of the essential grammatical distinctions cannot be made.

Systems for Full-Scale Analysis

The basis for several developments in automatic syntactic analysis is a Cocke-type parsing algorithm, which operates in a bottom-to-top mode using a table of binary context free phrase structure rules (for specific references, see Montgomery 1969). Although the Cocke algorithm is exhaustive and rapid, it has the disadvantage of requiring large amounts of storage and producing multiple analyses of sentences.

A well-known system for syntactic analysis of English is the Kuno-Oettinger Syntactic Analyzer (Kuno & Oettinger 1963),

which uses a context free phrase structure grammar of several thousand rules and a top-to-bottom analytical procedure based on a pushdown store. It is presumably more economical of storage than a Cocke-type analyzer; however, it also suffers from the disadvantage of generating multiple analyses of rather simple sentences. Moreover, it has a further characteristic disadvantage of top-down parsers: that is to say, it tests for applicability all rules having such non-unique initial symbols as SE, PRN, PRED,[7] only one of which will be relevant in the particular analysis of the given sentence. Thus the Cocke algorithm constructs all possible well-formed substrings with respect to the given string, while the Kuno-Oettinger Syntactic Analyzer essentially constructs all possible well-formed strings with respect to the particular grammar.

Another top-down syntactic analyzer, which is claimed to be adequate for an information retrieval system is a procedure for string decomposition of sentences (Sager 1968). This system, which is based on Harris' theory of string analysis, provides for the analysis of a sentence into its component strings, one of these being an elementary sentence (essentially a kernel) to which all other strings are joined directly or indirectly. Atomic strings are grammatical categories; however, these -- like all strings -- are further classed on the basis of how they can

be inserted into other strings, e.g. as left, right, or sentence adjuncts, replacement, conjunctional, or center strings. The grammar consists of a set of definitions or rewrite rules, each of which has an associated set of restrictions or well-formedness checks. The rules are therefore context sensitive, and the grammar accordingly acquires greater power, although it is less than a tenth of the size of the Kuno-Oettinger Syntactic Analyzer.

Syntactic analyzers which are more powerful than those based on phrase structure grammars are known as transformational parsers; they are capable of relating sentences such as (a) and (b):

(a) Smith wrote the report.

(b) The report was written by Smith.

The transformational approach is based on the assumption that each sentence has a "deep" structure underlying the "surface" structure which is its actual realization in writing or in speech.[8] A component consisting of phrase structure rules generates a base tree into which items from the lexicon are inserted to produce the deep structure representation. A component composed of transformational rules -- that is, rules which may adjoin, delete, or substitute items -- then operates on the deep structure to produce the surface realization of the sentence. Deep and surface structures are thus interrelated by an intricate series of

transformations, and sentences like (a) and (b) are related in that they are derived from the same deep structure, but have different transformational histories, (b) having undergone the passive transformation (and other relevant transformations).

The MITRE procedure for syntactic analysis is a system of this type (Zwicky, et. al., 1965). The initial step of the procedure is a dictionary lookup, which supplies for each item in the input string an appropriate set (possibly restricted to a single member) of "pre-trees" specifying the particular lexical and grammatical roles the item may assume. The first step of the analytic procedure is a surface-structure parsing of the input string as represented by the alternative combinations of pre-trees, using a context free phrase structure grammar. The surface trees are then mapped into potential base trees by reversing applicable transformational rules. Two checks are subsequently performed to validate the base trees. A test is first made to determine whether the base tree can be generated by the phrase structure component of the forward (generative) grammar; then all possible (forward) transformations are applied to the base tree, creating a new surface tree which must match the surface tree constructed as a result of the context free parsing.

Development of a recognition algorithm with an ability to relate sentences (a) and (b) is also being carried out by Petrick (1965). Like the MITRE strategy, Petrick's algorithm uses a

transformation reversal procedure, essentially converting a generativeor "forward" transformational grammar to a recognition or "reverse" grammar.

A type of syntactic analyzer having power equivalent to the above transformation parsers, but without specific transformational apparatus is currently under development by Thorne, Bratley, and Dewar (1968), Bobrow and Fraser (1969), and Woods (1970).[9] The basic concept is that of a finite-state transition graph; however, the model is augmented by certain features to provide power equivalent to that of transformational parsers. Specifically, the "augmentation" consists in the addition of two features to each arc of a non-deterministic finite-state transition graph: 1) a condition which requires satisfaction in order that the particular arc may be followed; 2) a set of "structure-building" operations to be performed if the arc is followed. The first feature provides the recursive capability of the model, since it involves the application of non-terminal symbols or state names to the graph. When a non-terminal symbol is encountered, the state at the end of the arc is saved on a pushdown store and control passes to the state which is the label on the arc -- essentially a subroutine call to a transition graph of the given name to determine whether the named non-terminal construction, say "noun phrase", is present. The second

feature of an "augmented" transition network provides for the construction of a partial structural description in a set of registers the contents of which are changed by the "structure-building operations"on each arc. (Registers may also hold "flags" which can be interrogated by the conditions on the arc.) The set of "structure-building" operations can produce transformational deep structure descriptions or structural descriptions appropriate to other theoretical frameworks.

Transformational and augmented transition network parsers are capable of relating sentences (a) and (b) above, which are derived from the same deep structure through the application of different transformational rules. However, none of these systems can relate sentences (a) and (b) to (c), a relation which is crucial for information science:

(c) Smith prepared the report.

The following two sections discuss some automated approaches to the solution of this problem.

Automated Semantic Analysis

In information science, content analysis has been largely limited to semantic analysis, which has generally been effected through some indexing vocabulary or system of content categorization. Some systems have been based on analysis of the relations between content terms and the specification of these through use

of roles and links, modifiers, and so forth; but such developments have been the exception rather than the rule. Thus there is little precedent for syntactic analysis and a correspondingly minimal motivation for attempting the automation of syntactic analysis.

In the case of semantic analysis, however, there has been a great deal of interest in automated processing, although efforts have been largely limited to various types of automated statistical analysis. Some of the more promising developments of this type involve statistical analysis of term associations within a particular indexing vocabulary. Jones, Curtice, Giuliano, and Sherry (1967) discuss experiments involving term associations based on co-occurrence in a coordinate indexing system, which also constitutes an initial step in associative experiments reported by Sparck-Jones and Needham (1968). In both cases, association is defined by co-occurrence of index terms (called "properties" by the latter authors) over a particular document collection -- in the case of Jones, et. al., a large collection, including some 100,000 documents and 18,000 terms; in the case of Sparck-Jones and Needham, 165 documents and 641 terms. The latter authors also generate "clumps" of maximally associated properties by various formulae, while the former use an arbitary cutoff point based on frequency of occurrence of individual terms to determine inclusion in association matrices.

Generally speaking, while these types of semantic analysis of entire document collections based on index terms are extremely useful, analysis of individual documents based simply on word frequency is not. As was noted above in the discussion of automated morphological analysis, some statistical analyses have even failed to regard singular and plural forms as occurrences of the same word. Others make some attempt to define a word in terms of a stem, but still neglect the basic semantic unit -- which is a concept, rather than a stem or a unique character string. Information on word frequencies is invaluable as a form of feedback data to a content analysis system (as I mentioned in discussing the concept of an information system), but it does not provide a semantic analysis of a document.

One of the few examples of a highly complex system for automated semantic analysis which includes no syntactic analysis is that of Wilks (1968). Using a dictionary in which each sense of a word is characterized by a "semantic formula" developed from a set of 52 semantic primitives (such as part-whole, causation, etc.), Wilks attempts to derive the semantic content of a paragraph of text. The sentences of the paragraph are first segmented into "fragments" which are concatenated for the entire paragraph across sentence boundaries. After

dictionary lookup, each fragment has a number of possible combinations of word senses for the various words of the fragment. These combinations or "frames" are then rewritten in a standard form called a "template", and tested for internal and external semantic compatibility by a series of complex operations, the ultimate objective being the production of a single string of templates representing the semantic content of the given paragraph.

Another system for replacing text words with conceptual categories is described by Laffal (1969). In contrast to Wilks' elaborate procedure, Laffal's is a simple direct substitution program; it does not examine the relationships between concepts, nor does it attempt to resolve ambiguities.

Automated Approaches Combining Syntactic and Semantic Analysis

An experimental model which has been under development for some time is Salton's SMART (1968). The model includes a number of different subprograms for document processing, query analysis (a query is treated as a minimal document), and retrieval in order to provide a capability for simulating different types of IR systems. When the available set of syntactic and semantic subcomponents are used, the input words are first processed through a suffix stripping algorithm, and the word stems are

then looked up in a thesaurus to obtain concept group numbers. Concepts which co-occur above a specified frequency in document sentences are called "statistical phrases"; the relations between the concepts of a statistical phrase may be verified by a syntactic analysis of the sentence. The syntactic phrases are then matched against appropriate "criterion trees", which specify concept numbers and permissible syntactic relations between the given concepts.

Salton has stated, however, that he considers syntactically determined phrases (as opposed to statistical phrases) too specific for document retrieval. There are several grounds for questioning the validity of this generalization. First, the syntactic analyzer used in the SMART system is the Kuno-Oettinger Syntactic Analyzer, the defects of which were noted above. Among these is the production of a multitude of different analyses for rather simple sentences, only one analysis being correct in the given case. In order to preserve some semblance of operating efficiency, the Kuno-Oettinger Syntactic Analyzer is not permitted to analyze indefinitely in the SMART system, but is halted after producing a single analysis. All things considered, this first analysis is hardly likely to be the correct one, and thus may fail the "criterion tree" test.

A further difficulty with Salton's assessment of value of syntactic analysis consists in the completeness of the specification of the "criterion trees" themselves. These trees function as templates for defining the set of syntactic relations which may link a particular phrase or term pair. It is clearly not a simple task to define all possible acceptable syntactic combinations of term pairs for a vocabulary of any size; consequently, it is probable that some acceptable combinations may not be specified in the "criterion tree" dictionary, causing rejection of valid phrases.

A final objection to Salton's assertion may be raised along the lines suggested in Footnote 3. This discussion involves the function of syntactic analysis in large data bases where the content of the data base is homogeneous. In such a situation, it would appear that document content and user requirements must be quite rigidly specified to achieve an acceptable degree of precision. For example, assuming a data base containing information on international finance, there must be some way of specifying relations such as donor-recipient, export-import, and the like, since content representations and search prescriptions in the form of Boolean combinations of terms would clearly be unsatisfactory.

The SMART system is unique in that it has been developed to serve as an experimental model for testing different approaches -- mainly statistically based -- to the construction of information system components. Other automated approaches combining syntactic and semantic analysis are described below under text processing systems and question-answering and fact retrieval models (for a discussion of the implications of this sub-classification, see Footnote 3).

Text Processing Systems

A model which has been under development for several years is Von Glasersfeld's "Correlational Grammar" -- an empirical approach based on the notion that syntax and semantics are not separable. Rather, they interact to produce a network of "correlational structures", which are pairs of constituents connected by a "correlator". Thus items are classified in terms of their roles in "correlational structures" (which may be syntactic or semantic) in contradistinction to the traditional view, according to which syntax specifies relations between items classified in terms of grammatical categories. To date, some 350 correlators have been developed, and correlational analysis begins by looking up each input word in a master table of correlations to determine which correlators are applicable. A process of "reclassification" is then initiated to rewrite

constituent pairs as single correlations.

An approach which also integrates syntactic and semantic categories is described by Noël (1966, 1970); however, his proposals are novel in several respects. First, his concept of semantic analysis specifically relates linguistic theory and documentation in terms of a metatheory which he defines. (This aspect of his work is more appropriately discussed below in the section entitled "Syntax versus Semantics".) To demonstrate the implications of his metatheory, Noël conducted an automated indexing experiment using an analytical procedure which constitutes a second novel aspect of his work. This procedure for text analysis is based on the concept of "shrinking" or reducing text to a single content expression which is equivalent to an entry (or entries) in a particular subject classification scheme. A third novel aspect of Noël's approach is that the text is considered as a discourse structure; periods are ignored and a text is treated as a set of conjoined sentences which are reduced to a single content expression.

The reduction operation is carried out by the successive application of several sets of rules for concatenating text elements, using a variant of the Sager string analysis procedure described above in the section on automated syntactic analysis. As noted above, the string analysis procedure specifies for each

rewrite rule a set of restrictions or well-formedness tests involving subcategorization features of the constituents. A "restrictionless" variant of the analytical procedure has also been proposed, where the need for well-formedness tests is obviated by developing additional string categories and rewrite rules; it is this version of string analysis which is utilized by Noël. It should be noted, however, that the Harris/Sager concept of "center string" is rejected as "semantically irrelevant", since it does not correspond in the majority of cases to the semantic units postulated by Noël.

The data base used in the automated indexing experiment consists of 50 document abstracts in the field of information science, and the indexing vocabulary is a concordance of several information science classification schemata prepared by Gardin. The analytical procedure relates the surface structures of the abstracts (the "object-language") to the surface structure of the classification entries (the "metalanguage"). The deep structures appear to be relational statements along the lines of Fillmore's role relations (1968, 1969) and the syntagmatic and paradigmatic relations of SYNTOL (Cros, Gardin & Levy 1964).[10] In an earlier discussion of the experiment (1966), the relational statements are the nodes of a semantic network which serves to

relate the text to the classification scheme.

The concept of a semantic network is elaborated by Quillian in his discussion of the "Teachable Language Comprehender" (TLC) (1969). The nodes in the network are specified as "units" and "properties", which Quillian equates with the logical concepts of argument and predicate, respectively. Units represent object, events and concepts, while properties represent relations. Properties may be conventional attribute value relations, or relations such as verb object. The first element of a unit is a pointer to another unit which is a superset of the original unit; other elements are pointers to the properties which mark the original unit as a particular subset of the given superset. Nodes are thus defined in terms of other nodes to which they point.

In analyzing an input text, each word is first looked up in a natural language dictionary external to the semantic memory. For each sense of a word, a dictionary entry contains a pointer to the particular unit in the memory which corresponds to the word sense. The process continues, attempting to define the newly created node by filling in pointers to appropriate supersets and properties. A search of the contiguous items in the input string is first undertaken, and if these fail to match any

properties specified in the candidate unit, the search ascends the superset hierarchy until an acceptable intersection is found. For example, if the input string is "lawyer's client" and the unit defining client contains the property "employ/professional", an attempted match on the attribute "employ" will fail, but the search on the value "lawyer" will succeed, since "lawyer" is a subset of "professional" in the semantic network. In Quillian's original TLC design, the validity of the connection is verified by syntactic well-formedness tests; however, in future stages of development, a transitional network parser will be used for syntactic analysis.

A device for semantic representation which is related to that of the network is the thesaurus. In a substantive article on thesaurus construction, Chernyj (1968) distinguishes three types of thesauri, which he labels "linguistic", "statistical", and "normative". Statistical thesauri are created by techniques such as those discussed above under automated semantic analysis. Normative thesauri are those that organize descriptors -- these are the main topic of his article, which includes a discussion of paradigmatic relations to be represented, methods for selecting descriptors based on statistical text analysis, and a detailed discussion on the construction of

thesaurus entries (including a flow chart). A linguistic thesaurus, on the other hand, contains natural language words rather than descriptors -- these natural language entries are selected through content analysis of text and constitute a system by virtue of their relation to a previously developed classification. The three following text processing concepts are concerned with different aspects of the design and development of such a thesaurus.

In proposing a design for automated text analysis, which he sees as a potential unifying factor for the theories of linguistics and documentation, [11] Petöfi (1969) is concerned with specifying the type of information a thesaurus should contain. The thesaurus which is the basic component of his system design will incorporate linguistic information -- i.e., information such as that included in the lexicon of a formal grammar, specifications of phonological, morphological, and lexical information -- as well as the "encyclopedic" information represented in a documentation thesaurus. The concepts of the thesaurus are linked by "logical semantic" relations, while the lexical units are linked by "linguistic semantic" relations; both sets of relations must be defined, as well as the interrelations of the "logical" and "linguistic" semantic systems.

In outlining a system concept for processing medical English, Pratt and Pacak (1969) are concerned with the problem of developing a "linguistic" thesaurus in Chernyj's sense from an existing "normative" thesaurus -- in this case the "Systematized Nomenclature of Pathology" (SNOP). In SNOP, terms -- which are mainly noun phrases of various types -- are uniquely assigned to one of four semantic categories basic to pathology: these are topography, morphology, etiology, and function. Within these categories, terms are hierarchically organized. For an automated system, however, it is clearly necessary that all acceptable natural language representations of a term must be available to the analytical algorithm. In order to achieve this goal, the authors suggest transformational rules, but the exact use they intend to make of such rules is not obvious. In any case, it would appear that the well-known idiosyncracies of English nominalization would preclude the possibility of automatically expanding all entries of the same type by a single set of transformational rules.

One of the major difficulties in attempting to automate content analysis is the lack of a theory of language which provides for precise specification of syntactic relations, let alone the relations which Petőfi calls "linguistic semantic"

and "logical semantic" and the interrelations of all three. Therefore, in designing a content analysis module for a large scale text processing system, the author and several colleagues adopted an essentially unrestricted concept for representing the semantic structure of the particular universe of discourse (Montgomery, Worthy, and Reitz 1968). We assume that it is more realistic to begin with an unrestricted representation and introduce restrictions as necessary rather than to construct at the outset a rigidly specified structure which will in the long run prove inadequate to cope with the richness of natural language syntax and semantics. We thus developed the concept of a natural language thesaurus for the identification and representation of semantic structure. This thesaurus allows identification of concepts as they appear in text -- i.e., as natural language words and phrases of any length or internal construction -- in Chernyj's terms, a "linguistic" thesaurus as opposed to a "normative" thesaurus, concepts are represented as strings of n elements, where an element may be a single word, a phrase of n words, or a set of n synonymous words and/or phrases. Obligatorily associated with each element are grammar codes expressing word class and subcategorization information; optionally associated with each element are concept codes which

relate the natural language words and phrases to the subject classification scheme describing the particular universe of discourse and search parameters for the identification algorithm.

There are three formats in which the thesaurus can be arranged: the thesaurus format, the dictionary or search format, and the wordmap. In the thesaurus and dictionary formats, the original structures were distinct trees, the first being organized in terms of the conceptual hierarchy embodied in the classification scheme, and the second interms of decision trees representing optimal search strategies. As the natural language thesaurus was expanded, however, many nodes in both types of trees were defined as pointers to content-bearing nodes in other trees to avoid redundant specification of concept elements. Thus the sets of distinct trees gradually evolved into a net structure where the nodes are either concept elements (as described above) or pointers to other nodes, the edges representing conjunction relations in the case of the dictionary and relations of inclusion, set membership, and conjunction in the thesaurus. In the third format, which provides an index or map or word occurrences within the thesaurus, the organization

resembles that of Quillian's TLC, in that each word is defined in terms of (i.e. points to) all its associations in the thesaurus (all its possible senses in the given universe of discourse).

The natural language thesaurus is the keystone of a text processing system developed for an automated indexing application. The content analysis subcomponent also includes a version of the Cocke algorithm which tests syntactic well-formedness of strings of concept elements recognized by the indexing (concept identification) algorithm.

Research Models for Question-Answering and Fact Retrieval

Since Simmons (1970) provides a detailed treatment of these models, this discussion will be limited to those which are recent and which have particular relevance for the topic of the final section.

A classic among these is the"Protosynthex III" model described by Simmons, Burger, and Schwarcz (1968). This system operates in the following manner. The words of input statements and questions are first looked up in a dictionary or lexicon in which the various possible senses of each word are associated with information on grammatical category, subcategorization specifications, and concept codes representing the set of semantic

classes which are supersets of the given word sense. A Cocke-type syntactic analyzer which combines the standard phrase structure rules with transformations is used to analyze the input string into concept-relation-concept "triples" -- the type of deep structure characterizing the formal language (in the sense of Figure 2 above) of the Protosynthex system. As each pair of constituents is transformed into a conceptual triple, the resulting constitute is checked for semantic well-formedness against a table of "semantic event forms". The latter are also triples; however, the concepts are semantic class terms -- the supersets of the concepts in the triple derived from the input string. As distinguished from the following four systems in which the semantic representations of input strings are procedures for operation on a data base, the conceptual triples are the basic data structures of the information store.

Protosynthex is a fairly elaborate model in the sense that it provides some sort of capability for most of the functions of the information system components specified in Figures 2 and 3 above. It accepts natural language statements and questions, transforms them into conceptual triples constituting the formal language of the system, performs searches involving either direct lookup or the application of deductive inference rules,

and generates answers which closely approximate natural language statements using a "forward" grammar (the inverse of the recognition grammar described above).

The only two question-answering models which take into account the information utilization component represented in Figure 1 are CONVERSE (Kellogg 1968) and REL (Thompson et al 1969, Dostert and Thompson 1970). In these models, the information utilization component of Figure 1 is in effect superimposed upon the content analysis component, since the translation between the natural language input and the formal language of the system is user-defined. Both systems are syntax-directed compilers which accept as input a user-defined subset of natural English, converting the input strings into statements in a formal language. These statements are then interpreted by the system as search and storage procedures, according to a set of category definitions specified by the user and the set of grammar rules which manipulate the category definitions.

The CONVERSE analysis procedure uses a dictionary which specifies syntactic categories and subcategorization, as well as semantic features and selection restrictions of the type proposed in Katz (1966). As the analysis of the input sentence proceeds, the semantic features of each governor are checked

against the selection restrictions specified in the entry for the dependent to determine whether the combination is semantically valid. In cases of syntactic ambiguity -- e.g. determining the correct constituent structure for a series of prepositional phrases -- the most probable syntactic structure is first selected and semantic interpretation of the resulting statements into the formal procedural language is then attempted. If this fails, control is returned to the syntactic analyzer, and an alternative syntactic interpretation is assigned and tested in the same way.

In contrast to CONVERSE, the dictionary of the REL (Rapidly Extensible Language) system does not contain any syntactic or semantic information other than the user's definition of the lexical item in terms of one of the "REL English" categories of name, relation, number, verb, time modifier, or relative clause. There is thus no means of verifying semantic validity of the syntactic constitutes other than by reference to the data base; a syntactic constitute is semantically valid if the associated semantic routine is a legitimate operation on the data base. For this reason -- as well as because the order in which the semantic routines are executed is preferably determined by the analysis of the entire sentence -- tests for semantic wellformedness

are not performed as each constitute is formed during syntactic analysis, but are deferred until analysis of the sentence is complete (unless the user specifically opts for the other alternative.) Syntactic ambiguities are controlled by use of syntactic features of various types, e.g., number and type of modification in noun phrases, tense and voice in verbs.

Both REL and CONVERSE can analyze facts and answer questions involving relations of set membership and inclusion, as well as other binary relations (e.g. location), and various combinations of these. REL also provides for specification of time modification and interprets verb tense, aspect, and the possessive case of nouns. Although CONVERSE does not include these capabilities, it has a richer system of semantic features and selection rules, as described above. In both cases, it is obvious that the indicated capabilities will ultimately be necessary in systems of this type; thus the deficiencies are characteristic of a particular level of development of the model.

Woods (1968) has designed a model which interposes a formal query language between the semantic interpreter and the retrieval component. Unlike Kellogg's model, in which syntactic and semantic analysis are to the extent

possible effected simultaneously, Woods assumes a Chomsky-type deep structure representation as input to the semantic interpreter. The model has a set of semantic primitives, which are predicates, functions, and commands appropriate to a U. S. Airlines Guide data base; these are interpreted computationally, as procedures to be carried out by the retrieval component. Two types of deep structure nodes--S nodes and NP nodes--are processed by the semantic interpreter, which uses a series of "templates" to verify syntactic and semantic wellformedness in translating the statement into the query language.

Culicover et al (1969) describe a restricted transformational grammar and semantic interpreter designed to answer questions about the content of a library. The semantic interpreter is similar to that of Woods in principle, but is less elaborate. The first processing step in going from input to retrieval is a dictionary lookup, which supplies grammatical labels for the input string. The string of grammar codes is then input to the "Reductions Analysis" routine - a set of ordered rules which bracket well-formed substrings and normalize certain types of structures by applying transformations which rewrite the structures or insert dummy symbols. The

objective of this operation is to restrict the number of possible deep structures or trees which are input to the semantic interpreter, in order to simplify semantic processing. The latter operation maps the deep structure trees into commands which are passed to the information system.

In contrast to the procedurally oriented semantic representations in the models of Kellogg, Thompson, Culicover, and Woods, the following question-answering and fact retrieval models utilize semantic representations which are data structures--specifically n-ary relational statements.

A classic system of this type is the Relational Data File developed by Levien and Maron (1967). The structures of the data base are elementary relational sentences consisting of a one or a two-place predicate and its associated arguments. Elementary sentences involving properties (one-place predicates) are represented by set membership relations, e.g., 'x is a book' (x is a member of the class 'books'); all other relations are represented by two-place predicates. Each argument may also be a pointer to another relational statement, as in 'Rachel wrote y', where y is a pointer to the statement 'y is a book'. The file currently contains some 70,000

relational sentences dealing with bibliographic data in the field of cybernetics. Specially designed bibliographic data forms were filled out by clerical personnel as a first step in compiling the data base. These data were then converted to machine-readable form and processed by a series of programs which used the information provided by the data formats to generate relational sentences.

In order to retrieve information from the file, requests are formulated in the INFEREX programming language which essentially translates the request into a formula of the predicate calculus. INFFREX permits the user to specify rules of inference for retrieval of data. Kuhns (1969) describes an algorithm for translating natural language queries into the symbolic language of the system, and discusses the implications of the extensional and intensional aspects of meaning for such a procedure. The extension of a two-place predicate is the set of ordered pairs of arguments which stand in that relation, while the intension refers to the set of meaning postulates giving the interrelations between predicates. Thus, the most difficult aspect of natural language to logic translation is the representation of the rich variety of natural language relations - e.g., synonymy, modality, time, quantification - in terms of a set of meaning postulates.

Other research directed toward the translation of natural language into a formal logical representation is reported by Bohnert and Backer (1966), Williams (1966), Poducheva (1968), and Coles (1969). The project described by the last of these authors represents the latest stage of development of a research effort integrating the question-answering models developed by Green and Raphael (1968) with a natural language to symbolic logic translator designed by Coles in order to communicate with a computer-controlled mobile robot. Coles' design involves a syntax-directed approach to the translation of natural language into a formal language, as in the REL and CONVERSE systems described above. In this effort, however, the formal language does not specify a set of procedures to be executed on the data base, but rather a set of statements in predicate calculus notation. These statements are passed to the inferential component, which evaluates them in terms of a set of axioms defining the robot's environment and a technique for proving theorems by refutation (Green and Raphael 1968). This method involves the treatment of a query as a postulated theorem, exploiting the notion that a theorem follows from its axioms by attempting to construct a model that is consistent both with the axioms and the negation of the postulated theorem.

If this attempt fails, the truth of the theorem is proved. The robot's natural language response to the question or command is then produced accordingly, using a generative analog of the natural language recognition grammar and, in some cases, incorporating information contained in the axioms defining the robot's environment. Requests for information about the environment and commands to perform tasks are subsequently passed to a subsystem controlling the robot for execution.

Shapiro and Woodmansee(1969) describe a question-answering model in which the nodes of the network are terms (e.g., x, y) of binary relational statements (xRy), the relations constituting labels on directed edges (labels are also nodes, allowing the storage of information about the relations themselves within the net structure). A capability for recursive definition of relations is included, as well as a facility for controlling the question-answering strategy by limiting the search to particular subclasses of all relations represented in the net. Logic is essentially user-defined since the user specifies the axioms (the basic set of binary relations) and rules of inference (the recursive definitions of relations).

An extension of this model is the MENS model, which allows xRy statements as nodes in the semantic network. The similarity of the MENS data structures to the concept-relation-concept triples of Protosynthex and to the elementary sentences of the Relational Data File is clear, and, as in the latter models, the terms of a relation may themselves be relational statements, allowing for multiply nested representations of n-ary relations. A version of MENS which can represent complex relational structures without multiple nesting is also under development (Kay/Su 1970). This model--unlike MENS and its parent system--accepts natural language input; it thus includes a sophisticated morphological analyzer (Martins 1970) and a powerful parser based on unrestricted rewrite rules (Kaplan 1970).[12]

A less formal approach is taken by Schank and Tesler (1969), who also use a net structure data representation. Their "Conceptual Dependency" parser analyzes natural language in terms of network of concepts (or unique word senses) interrelated by semantic dependency links. Dependency is defined in terms of two criteria: the dependent concept must in some sense provide additional information on its governor; and the governor must be necessary to the understanding of the dependent concept.

The conceptual structures are mapped into sentences by a set of "Realization Rules," which are reversed to provide an analytical capability. An interesting feature of this system design is an attempt to build in a quasi self-organizing system for testing semantic wellformedness by maintaining a list of "experiences" (conceptual connections previously input to the parser). If a given construct has not previously been presented to the system, the user is interrogated as to the acceptability of the connection.[13]

A design proposed by Becker (1969) integrates in a single model of semantic memory features characterizing the different types of models discussed above. The semantic memory is a net structure, where nodes represent concepts - either atomic, such as the name of a particular individual, or complex, denoting sets of other nodes ("hero") or relations involving higher order constructs ("give"). The next higher order construct in Becker's model is a "kernel", an ordered n-tuple of nodes representing a predicate (designated by the initial node) and its arguments. Kernels are utilized to construct "situations", which are interpreted as conjunctions of statements expressed by the kernels. "Rules" are ordered pairs of

situations, where the situation comprising the right half of the rule is in some sense a consequence of the situation comprising its left half. The exact nature of the consequence is unspecified; it is interpreted by the particular processes operating on the given data structure. Thus Becker states that a particular rule may operate in separate instances as a predicate calculus formula or a procedural rule of the "pattern-operation" type. In this sense, the semantic memory operates simultaneously as a data structure (cf. the models of Simmons and Quillian discussed above) and as a procedural language (cf. the models of of Kellogg, Thompson, Woods, and Culicover discussed above). The formalism is also similar to the predicate calculus notation used in the models discussed in the preceding paragraphs. Becker's model is thus an interesting attempt to integrate several concepts of semantic representation; moreover, elementary cognitive subprocesses of analogy and generalization which the model is designed to simulate are evidently significant in automated "understanding" of natural language, although computer implementation on any realistic basis appears very remote.

All of the models discussed above exhibit some features which constitute promising approaches to the automated 'understanding' of natural language text. However, many of these models are only at the system concept stage of development. Of those which have been implemented on a computer--with the exception of the system described in Montgomery et al (1968)--the implementations have been experimental and the system components quite limited in scope. The dictionaries and other files typically contain very few entries, (the RAND Relational Data File and the thesaurus used by Montgomery are exceptions to this generalization), and some models have actually processed only a few facts, propositions, or sentences of text. The concept of an automated system for understanding natural language is necessarily complex--as are these models; however, the complexities of the interactions between subcomponents can scarcely be approximated on such a small scale.

In order to be useful in any kind of operational context, a system for automated understanding of natural language must be designed to accept high volume input and must inevitably include large dictionaries and complex components for syntactic and semantic interpretation. It can be anticipated that the system parameter of sheer

volume will introduce problems which cannot be predicted within the limited framework of a small research model. On the other hand, many of the problems revealed in the context of these models are not those which will be crucial in a large scale implementation, and for those problems which are critical, the solutions presented in the model are not likely to be valid within an operational context.

One example of a crucial problem area is the general (again, the system described in Montgomery et al is an exception) lack of a capability to modify and improve system performance through various types of feedback data. As discussed above under the specific models, most components do not lend themselves easily to modification, let alone include capabilities for collecting feedback information.

In terms of the information systems concept presented in Figure 1, it is clear that the various feedback loops are generally missing. Moreover, the component for content analysis and representation has been elaborated at the expense of the other components in almost all models. Considering the systems presented in the first section of this paper, as well as the models discussed in this section, we find that approaches to natural language

data processing have been for the most part limited to large scale data management systems on the one hand, and small scale models featuring elaborate components for content analysis and representation on the other. None of these approaches reflects a real concern with the user;[14] in the case of the data management system, the more complex functions of an information system--e.g., content analysis and representation--must be performed by the user, whereas the research models may include intricate features which have little or no value in an operational context, while lacking other more essential capabilities. The data bases on which the large volume types of systems operate presumably have some informative function in the real world; however, the characteristically small data bases associated with question-answering and fact retrieval models have no practical function, and in all probability, bear little resemblance to fact files maintained by particular classes of actual users. A further deficiency of most systems of both types is the lack of a super-system to monitor performance and provide feedback data for improvement of system components and dynamic adaptation to changing requirements.

It is clear that the computational linguistic models complement the large-scale data management systems in the sense

that the weaknesses of the one are offset by the strengths of the other. What remains to be seen is whether these types of approaches can be integrated in some meaningful way in order to exploit the strong points of both. This question is explored in the next section, which examines the state of our knowledge about language in the context of recent developments in linguistic theory.

SYNTAX VERSUS SEMANTICS

From the material presented in the preceding section, it is evident that the most solid achievements in computational linguistics involve syntax. To state that this is attributable to the more elusive nature of meaning is almost a truism; however, it is also attributable in no small measure to the syntax-based orientation which has characterized linguistic theory since the publication of Syntactic Structures (Chomsky 1957).

Due to the linguistic theoretician's preoccupation with syntax and the formal properties of grammar (Chomsky 1963), efforts in natural language data processing were devoted almost exclusively to parsing strategies and background research was concentrated on automata theory and mathematical linguistics. Many of these research efforts contributed very little to the study of natural

as opposed to artificial languages. An interesting evaluation of the work of several such theoreticians is due to Kiefer (1968). In treating the set theoretical model of Kulagina and the generative model of Šaumjan (as well as aspects of the work of other East European mathematical linguists), Kiefer proposes several criteria, one of which raises the question of whether the particular mathematical model can be considered at all relevant linguistically. The fact that Kiefer finds it necessary to include such a criterion is indicative of the rather dubious utility of some mathematical models proposed for the explication of natural language. A somewhat different approach is taken by Harris (1968), who reformulates a previously developed linguistic model in mathematical terms.

This is not to disparage the genuine contributions of automata theory and mathematical linguistics to natural language data processing and the explication of a theory language. With respect to the latter, the contributions of Sakai (1968) and Zadeh (1970)--to be discussed below--are cases in point. A recent example of the former is the parsing algorithm described by Earley (1970), which essentially combines features of top-down and bottom-up

parsing strategies (see above under Automated Syntactic Analysis) to produce an efficient context free parser. Rather the issue is that research efforts in the theory of automata and mathematical linguistics in general reenforced the centrality of syntax in linguistic theory, ignoring the more basic role of semantics. As Varga has noted, in discussing the artificial compartmentalization of syntactic and semantic components in automated language analysis:

> " ...in human understanding, semantic processing of (hypothetically) isolated parts of the structure occurs before their definitive position in the system as a whole is ascertained...It seems that we would not be wrong to assert that the process of understanding can be represented as a sequence of specific semantic transformations which modify and interconnect the separate semantic blocks." (1968, p. 21).

The implications for linguistic theory of the role of semantics in the process of human understanding have, however, gone unnoticed by most linguists until recently. A notable exception to this generalization are the many Soviet linguists who are active in the area of lexicology;

the Soviet linguistic tradition is unique in that lexicology and related semantic studies have been objects of linguistic investigation equally as respectable as phonology and grammar. Ample evidence of the productivity of Soviet lexicologists is presented in the bibliography accompanying Weinreich's survey of the field, which lists over 250 items (Weinreich 1963).[15]

Some American linguists have recently begun to debate the issue of a syntax-based linguistic theory versus one which is semantics-based. The latter group would redefine the relations between components of a grammar, such that the creative element is the semantic rather than the syntactic component. (The original basis of the controversy is summarized in Montgomery 1969; for specific references see Lakoff 1968, McCawley 1968, Fillmore 1968.) More crucially, the feeling is that -- due to the primacy of syntax -- previously defined linguistic "universals" are oriented too much toward specific features of particular languages to account for the common features of all languages; Chomsky's notion of "deep structure" as the basic level of the syntactic component is not sufficiently removed from the idiosyncratic surface structure of a given language to reflect the fundamental categories and relations which are in fact universal.

The details of the issues in the controversy now labeled 'Lexicalist' versus 'Transformationalist' are too transitory to merit discussion; however, although the theoretical arguments tend to vary considerably, the real significance of the debate from the point of view of information scientists resides in the linguists' increasing concern with the fundamental problem of representing lexical and sentential meaning. Evidence of this concern is seen in the work of Bierwisch (1967), Gruber (1968), Fillmore (1969), Lyons (1968), and Leech (1970), among others. The novel interest in semantics is reflected in the work of computational linguists, as a selection of papers from the 1969 International Conference on Computational Linguistics shows (see Bellert 1969, Karttunen 1969, Rouault 1969, Schwarcz 1969, Vasiliu 1969, Vauquois et al 1969).

Perhaps the most encouraging note is the emergence of certain fundamental principles to which a majority of these researchers are committed. Central among these is the notion of the predicate as pivotal in semantic and syntactic analysis. As distinguished from the notion of a predicate in traditional grammar, where the appropriateness of the label 'predicate' is determined by the surface structure of a sentence and the notion 'subject' is of equal or greater significance, the term 'predicate' in

this context designates any relation holding between two or more entities (its arguments in the logical sense) or any property of an entity. This common thread runs through the work of the investigators mentioned above, as well as that of Apresjan, Zholkovskij, and Mel'chuk (1968), Garvin et al (1967) and earlier publications on the 'fulcrum' approach), the dependency grammars of Tesniere (1959), Hays (1964), and Robinson (1967), the documentation language SYNTOL (Cros, Gardin, and Levy 1964), and obviously, the formalism of mathematical logic which has been widely used in computational models, as discussed in the preceding section.

Of the various formalisms for syntactic/semantic representation, that of Fillmore appears to have the most explanatory power, as well as the most explicit mechanism for relating the formal language to natural language sentences. Fillmore (1968) postulates that the basic structure of a sentence includes a 'proposition' and a 'modality' constituent. The notion of proposition designates a set of relations represented by a predicate (which may be verbal or nominal) and its arguments, which are noun phrases or propositions; the modality constituent specifies tense, mood, and negation. The novelty of Fillmore's approach resides in his concept of the arguments

of a proposition as an ordered set of deep structure 'case' categories representing the fundamental 'role' notions which reflect human judgments about events or situations --e.g., who caused the event, who experienced it, what might have undergone a change of state. Specifically, these notions of 'role' include the following:[16]

Agent (A) - the principal--generally animate--cause of an event or the instigator of an action;

Instrument (I) - the 'efficient' cause of an event, a physical instrument;

Experiencer (E) - one to whom the event happens, one who undergoes a psychological event, or receives the effect of an action;

Object (O) - the neutral role, the content of the experience;

Source (S) - location or state of origin;

Goal (G) - final state or location.

Some surface structure realization of these roles are exemplified by the following sentences:

(d) Harvey broke the mirror with a hammer.
(A) (O) (I)

(e) Joe put the chairs against the wall.
(A) (O) (G)

(f) Anne gave the money to Claire.
(A) (O) (E)

Predicates may thus be classified in terms of 'case frames', rather than simply as 'n-place' predicates, as in symbolic logic; for example, the case frame for the predicate 'give' might be represented as [_____A O E]. In some instances, a role may be facultative. The case frame for 'break' and other similar verbs is [_____ (A) (I) O], where the parenthesized roles are facultative, as shown by the following sentences:

(g) Harvey broke the mirror with a hammer.
(A) (O) (I)

(h) The hammer broke the mirror.
(I) (O)

(i) The mirror broke.
(O)

Moreover, more than one role may be expressed by a particular argument. In the following sentence, the Source of the action and the Agent are shared by the noun phrase 'John':

(j) 'John threw a tomato at the actor'
(A) (O) (G)
(S)

The deep structure representation of such a sentence consists of a tree in which the noun phrases are dominated by nodes indicating their role relations. A set of rules for selection of subject, object, and prepositions, as well as additional related rules and the 'standard' set of transformational rules operate to produce the appropriate surface structure.

Returning to the notion of a natural language information system elaborated in Figures 1, 2, and 3, above--and in particular to Figure 1, the component for content analysis and representation--it is clear that the operations listed simply as 'translate' are at once the most essential and the most difficult to automate, since it is exactly these operations which involve 'understanding' the input text. As noted in the first section of this paper, the process of 'understanding' the content of natural language text involves identification of the concepts contained in the text and determination of the relations linking these concepts. In order to achieve this, there must exist a framework for specifying these concepts and their interrelations, which Petofi (1969) has classified into 'linguistic semantic' and 'logical semantic' (see preceding section).

In fact, much of the recent work in linguistics, as well as in computational linguistics, might be entitled 'In search of a formalism for content representation'. I suggest that Fillmore's 'case' grammar and underlying role notions provide a linguistically-based formalism for representing content in terms of relational statements which can accommodate both 'linguistic semantic' relations (e.g., all verbs having the case frame [_____(A) (I) O]) and 'logical semantic' relations--those which are not internal to the sentence (or derivable from the linguistic context), but are rather interpreted in terms of the 'encyclopedia' which is a speaker's knowledge of the world.

In addition to the 'logical semantic' relations discussed above in connection with the various computational linguistic models (e.g., set membership), Fillmore's notion of 'presupposition' and the interpretation of implication elaborated in Lyons (1968), Bellert (1969), and Leech (1970) are of particular significance in the explication of the process of 'understanding' natural language sentences.

later
In a/(1969) paper, Fillmore is concerned with a level of semantic description which is capable of characterizing the preconditions for appropriate use of a sentence. For

example, use of the simple imperative sentence 'Please shut the door' conveys implicit information dealing with the social and spatial relationship of speaker and addressee, the physical attitude of a particular door, and the desire of the speaker to change the physical attitude of the given door. In the cited article, Fillmore discusses a set of verbs involving judgments about situations and develops a notation for representing the meaning of these verbs and the 'presuppositions', or set of conditions which their use presupposes. The meaning statement and sets of presuppositions are defined in terms of the 'role structure' of the verb, which is described by Fillmore as 'analogous to' but 'distinct from' the role notions underlying 'case' grammar (1968).

The role structure specified in Fillmore (1969) includes the role concepts of Judge, Defendant, Situation, and the Affected individual, which are used as terms in a set of propositions and identity conditions to represent the content of the verb. Paraphrasing Fillmore's notation, the role structure of the verb 'accuse' involves the role concepts of Judge, Defendant, and Situation, and the meaning consists in a statement by the Judge to an addressee (who may coincide with the Defendant, i.e., 'you'), that

the Defendant is responsible for the situation, the presupposition being that the situation is bad. In discussing the 'verbs of judging' 'accuse', criticize', 'scold', 'blame', 'excuse' and 'justify', Fillmore notes the relationship between the situation parameters of 'badness' and responsibility for the situation, stating 'whenever one of these showed up in the description of the meaning, the other showed up in the statement of the presuppositions' (1969: 110). From this observation, it is apparent that the types of information contained in the 'meaning' statement and those contained in the presuppositions are not distinct; thus it follows that both can be represented by the same formalism--a point which is developed below.

Another important notion in semantic representation is that of implication. Leech (1970) defines a binary relational statement 'a·r·b' and stipulates:[17]

> 'An assertion a·r·b implies an assertion c·r·b if (the specifications being otherwise identical)
> (i) a logically includes c.
> (ii) the clusters [complex symbols] a and c are construed as if universally quantified.

Examples are:

"Children love apples" implies "Boys love apples".
"Men are mortal" implies "Postmen are mortal".
"I love fruit" implies "I love apples".'

(1970: 35)

This type of implication is called 'deductive' by Leech, who continues by presenting rules for 'inductive' implication, which--unlike deductive implication-- applies both to arguments and to predicates. Examples are:

'"John is eating peanuts" implies "John is eating nuts".'

'"Children ran down the street" implies "Children went down the street".'

Leech (1970: 36)

In concluding the section on logical implication, Leech notes a difficulty in dealing with attributes which are understood in terms of the entities to which they are applied:

'"A tall four-year old child lives next door" does not imply "A tall person lives next door".'

Leech (1970: 38)

For, as noted above, the 'logical semantic' relations, which include presupposition and implication, have to do with encyclopedic knowledge rather than with linguistic knowledge. It is perhaps for this reapon Fillmore states that the role concepts developed in his 'verbs of judging' paper are 'analogous to' but 'distinct from' the role notions underlying case grammar. However, I postulate that both notions are relatable in terms of a metalanguage specifying a particular universe of discourse (a subset of the encyclopedia) and that the deep structure relational statements of case grammar--enriched by some few of the devices discussed above in connection with the various computational linguistic models--are appropriate for representing both 'linguistic semantic' and 'logical semantic' relations.

These relational statements will thus consist of predicates and arguments, which may be represented by expressions designating entities or by pointers to other relational statements. Taking the earlier example, and assuming it has been 'understood' in terms of a metalanguage specifying a particular universe of discourse appropriate to it, 'Harvey broke the mirror with a hammer', 'Harvey', 'mirror', and 'hammer' would all be represented as objects. If the example is complicated by changing the

first argument to 'Harvey's father', that argument would be represented by a pointer to a relational statement of the type 'x is father of Harvey'.

The arguments (objects or relational statements) of a given relational statement are also arguments of other relational statements which specify the set of 'linguistic semantic' and 'logical semantic' relations in which a given object or relational statement participates. In the above example, Harvey[18] is simultaneously an argument of at least the following: (1) the particular relation described by predicate 'break'; (2) the primitive binary 'agent' relationship, where the second argument represents the proposition described in (1);[19] (3) the property given by a set membership relation--e.g., Harvey is a member of the class children.

Arguments which are pointers to other relational statements may designate presuppositions, as well as embedded sentences like 'Harvey's father'. Using Fillmore's 'criticize' example, (assuming once again that it has been 'understood' in terms of a metalanguage specifying a universe of discourse appropriate to it), the two presuppositions are that the 'Defendant' is responsible for the

situation and that the situation is factual, as in

John criticized Harry for writing the letter.

In terms of deep structure case relations, John is an argument of the predicate 'criticize', and is also an argument of the binary Agent relation, as in (2) above. Similarly, Harry is an argument of the predicate 'criticize' and of the binary Goal relation. Another argument is given by a pointer to a relational statement representing the embedded sentence 'Harry wrote the letter'; this argument is also an argument of a binary Object relation (in this case both arguments of the Object relation are themselves given through relational statements).

Extending this role structure to the notions advanced in Fillmore (1969), the argument of the predicate 'criticize' which is an argument of the Agent relation is also involved in a relational statement of the form 'x judge y',[20] where x is an Agent and y is a relational statement consisting of a one-place predicate 'bad', its argument being the relational statement representing the situation Harry wrote the letter, which enters into the Object relation as an argument of the predicate 'criticize'. The presupposition that the Defendant is responsible for the situation is thus represented by the identity of the Goal argument of the predicate 'criticize' and the Agent argument of the predicate 'wrote', the Defendant being synonymous with the Goal in this case. The presupposition that the

situation is factual is satisfied by the storage of the relational statement itself.

As a data structure for storage of these content representations, I suggest the concept of a network incorporating the features of several of the computational linguistic models discussed in the preceding section. The predicates and arguments of relational statements would be represented as nodes in the network, allowing node definitions in terms of other relational statements. Such a feature is necessary for predicates as well as arguments, in order to provide for specification of the 'linguistic semantic' relations conveyed by Fillmore's 'modality constituent' (e.g., tense) and for 'logical semantic' relations (e.g., Leech's 'inductive implication'). An index node points to each relational statement. Other pointers -- e.g., those specified in the above examples as arguments of particular predicates -- point to the index nodes of the relational statements they represent.

The structure of the metalanguage specifying the particular universe of discourse or subset of the 'encyclopedia' is thus represented as a network, where the basic concepts of the metalanguage are given by a set of nodes which are indexes to the complex of relational

statements defining these concepts and their interrelations. In terms of the computational models discussed in the preceding section, two alternative modes of accessing the network in text processing can be envisaged. One possibility consists in associating with the conceptual network a lexicon similar to the 'word map' format of the natural language thesaurus described in Montgomery et al (1968) or Quillian's TLC dictionary (1969). The lexicon provides grammatical information for a word in terms of each of its occurrences in the network and contains pointers to these. An alternative possibility consists in a frequency ordered dictionary of relational statements similar to the dictionary format generated for the natural language thesaurus described in Montgomery et al. Using this method, when the least frequently occurring element in some relational statement listed in the dictionary is encountered in text, content analysis of the contiguous portions of the text in terms of that relational statement (and other relevent statements) is initiated. In the following discussions, the first alternative is used.

As an example, assume that we are attempting to index automatically documents which belong to the universe of discourse described as 'computer science' using some appropriate metalanguage -- say, a subject classification

of the field, organized into the above-described net structure. If a document deals with computer science as an educational discipline in a university, it might contain a sentence such as the following:

(k) The WATFIVE compiler is mainly used by beginning students.

Assuming for the sake of argument that the first content word 'WATFIVE' is not represented by a distinct node in the network, the next word looked up by the indexing algorithm is 'compiler'. The lexicon will contain an entry specifying 'compiler' as an instrumental noun -- specifically, it instantiates the Instrument relation as one of the arguments of the predicate 'compile', where the other arguments represent the Object, Goal, and Agent roles. Thus, 'compile' can occur in the case frames exemplified in the following sentences: [21]

(l) The students compile their programs into an inter-
(A) (O)
mediate language with this compiler.
(G) (I)

(m) The students compiled their programs with the WATFIVE
(A) (O)
compiler.
(I)

(n) This compiler compiles programs into an intermediate
(A) (O)
language.
(G)

(o) The WATFIVE compiler compiled their programs.
(I) (O)

(p) Their programs compiled.
(O)

In addition to grammatical information, the lexical entry contains pointers to relational statements specifying the arguments which may occur as Instrument, Object, and Goal of the verb 'compile'. The only valid arguments[22] for the instrument relation are the term 'compiler', a hypernym for the software concept, the term 'computer' ('machine', etc.), a hypernym for the hardware concept, or the name of a particular computer system 'IBM 360', 'CDC 6600'), representing both hardware and software concepts. For the Object relation, the term 'program' is the only acceptable argument, while the arguments to the Goal relation must be the term 'language', or a synonym, such as 'code'. It is assumed these terms are further defined in terms of relational statements. 'Program' in the abstract sense ('$program_1$') is defined as a series of statements specifying a procedure and concretely ('$program_2$') as a realization of '$program_1$' in terms of a language, which presupposes the existence of that language. A language is in turn defined as involving a set of symbols, a set of rules for symbol combinations, and a set of rules for interpretation. The meaning of

'compile' can then be represented in terms of a relational statement such as 'translate (into) x, y',[20] where x (in the sense of 'program$_2$') is a series of statements coded in language A specifying a procedure and y is a series of statements in language B specifying the same procedure. The presuppositions are that x exists, that the series of statements comprising y will be larger than that comprising x (since the notion of compiling entails translating from a more powerful to a less powerful language), and that the pair of languages specified in x and y is unique to the particular instantiation of the translate notion.

In addition to this information, the content analysis of the input sentence requires an equivalence relation or a transformational operation[23] to relate the deep structure of that sentence to the structure represented in the lexical entry for 'compile'. This is necessary, since the input sentence (k) and the following variation on sentence (m), which is derived from the lexical entry for 'compile', are clearly equivalent:

(q) Beginning students mainly compile programs with the WATFIVE compiler.

This additional information allows the specification of the meaning of the unknown word 'WATFIVE' in terms of the definition for the predicate 'compile' and related statements described above, 'WATFIVE' being 'language A' of the x argument for the abstract predicate 'translate (into)'. Through

the latter relational statement, the input terms 'compiler' and 'WATFIVE' are also related to other notions whose meanings are specified in terms of the same statement -- e.g., 'interpret', 'assemble' -- and ultimately, to the more basic notions of computer systems, software, and hardware.

Concluding this rather lengthy illustrative example, it is illuminating to view the proposed concept for automatically 'understanding' natural language text in the context of Noël's metatheoretical proposals (1970).

LINGUISTICS AND INFORMATION SCIENCE: SUGGESTIONS FOR A METATHEORY

In proposing a semantic metatheory relating linguistic theory and documentation practice, Noel defines the three components of metalanguage, theory, and procedure, following Gardin (1969). The term 'metalanguage' specifies a 'public' metalanguage, such as a document classification system, as distinguished from the 'object language' represented by the documents, while 'theory' is an eclectic notion of current linguistic theory. The term 'procedure' designates an explicit system of rules which are based on the theory, and which serve to relate the metalanguage and object language. Thus, Noël's automated indexing experiment described in the preceding section was an attempt to define such a procedure, based on the linguistic theory of Harris, using a concordance of information science classifications as a metalanguage and 50 abstracts of information science documents as the object language data.

Although the approach proposed above is also aimed at integrating linguistic theory and information science techniques, it contrasts in some respects with Noël's metatheoretical concept as exemplified by his automated indexing experiment and as discussed in a portion of his dissertation (Noël 1970)[24]. The approach I propose is based on a somewhat different view of what constitutes the metalanguage specifying a given universe of discourse or subset of the encyclopedia. As noted above, Noël's definition of a metalanguage coincides with that of a document classification system; moreover, in parallel with the recent linguistic theoretical notion that language is characterized by a deep and a surface structure, Noël attributes the same characteristic to the metalanguage. He further postulates that deep and surface structures of both object language (natural language) and metalanguage are related in terms of the same theory (i.e., his eclectic notion of current linguistic theory). There are some difficulties with this proposal, however, since the written record of a document classification schema (with the exception of derived classifications such as KWIC) is not really parallel to the surface structure of the object language--the natural language sentences of a document. A classification schema is intended to classify, and, therefore, the language of the schema is mainly classificatory. the phrase 'non-numeric-programming languages' may be used to represent

the rather considerable extension of this set instead of listing the names of the individuals comprising it -- e.g., 'SNOBOL', 'LISP', 'PLI', etc. In other words, the metalanguage does not explicitly include all relevant terms in the object language, but the object language does include all terms in the metalanguage. Moreover, superset-subset (class inclusion) relations are usually explicitly given by the structure of the classification: 'non-numeric programming languages' is by one mode or another specified as a 'narrower term' with respect to 'programming languages'. Thus, some of the 'logical semantic' relations (specifically, those of 'implication', after Leech (1970)) are specified in the so-called 'surface structure' of the metalanguage, but not in the surface structure of the object language (i.e., natural language). Since the purpose of a metalanguage is the explication of an object language, the inclusion of such relations is wholly appropriate. However, this brings up the question of whether all relational information necessary for the explication of an object language (see the 'compiler' example elaborated above) is included in a metalanguage which is simply a document classification schema. Obviously, this is not the case: role notions and presuppositions are missing, among other things.

So a document classification schema is really not a metalanguage: it does not have the required explanatory power. On the other hand it obviously has a built-in explanatory structure which distinguishes it from the surface structure of the object language. These are my points of contention with Noel's metatheoretical proposals. But if a document classification schema is neither metalanguage nor object language, then exactly what is it, and where does it fit in, since the notion of document classification is clearly relevant in the content analysis of natural language, which provides the focus for integration of linguistic theory and information science practice.

A document classification schema is in fact an approximate representation of a metalanguage in an object language; for this reason, it exhibits some characteristics of both, but does not fully satisfy the criteria for one or the other. Its significance derives from its function as a vehicle for the expression of some of the fundamental notions of the metalanguage specifying the given universe of discourse. In and of itself, it is inadequate as a metalanguage; however, it is extremely important, because it provides a foundation for the construction of a metalanguage, and it is exactly this concept which has been lacking in the semantic investigations of linguistic theorists.

As linguists are discovering, semantic analysis requires the encyclopedia in order to account for the 'logical semantic' relations of presupposition and implication, as well as primitive notions. What has not yet been discovered is how to deal systematically with encyclopedic knowledge--which is where information science practice and document classification schemata come in. By this, I mean to suggest that it is possible to achieve a systematic approach to the explication of encyclopedic knowledge, using a document classification schema as a basis for isolating a subset of the encyclopedia within which 'logical semantic' relations can be defined as indicated by the 'compiler' example given above. The process of defining these relations will ultimately result in the development of a metalanguage specifying a particular subset of encyclopedia, which can then be used to interpret the corresponding object language (i.e., natural language sentences appropriate to the given universe of discourse). This is not to say the intellectual task will be trivial, nor that all problems of content analysis are necessarily resolvable in the foreseeable future - quite the contrary; however, this appears to be a realistic approach to a mind-boggling problem, which is more than can be said for attempting to deal with the encyclopedia virtually in its entirety, as most linguistic investigators seem to be doing.

Similarly, rather than search for a universal set of semantic primitives, a more realistic goal is to isolate primitive notions within a given subset of the encyclopedia. If a number of these subsets can be exhaustively specified, such specifications might provide some evidence to settle current speculations as to the existence of a universal set of primitives and their nature.[25]

To summarize: I essentially agree with Noël's (and Gardin's) definition of a metatheory relating linguistics and information science in terms of the three components metalanguage, theory, and procedure. However, my concept of a metalanguage involves use of a document classification schema as a basis for elaborating a metalanguage specifying a subset of the encyclopedia, rather than as a metalanguage in itself, as suggested by Noël. As indicated by the 'compiler' example above, my notion of a metalanguage involves 'logical semantic' relations and primitive notions represented as relational statements and organized into a net structure. The theoretical foundations of the metalanguage (Noël's theoretical component) are Fillmore's role notions and the concepts of relational logic embodied in several of the computational linguistic models discussed in the preceding section. Thus the procedure requires a parser with a

transformational capability -- e.g., Woods' augmented transition network parser (1970) or the unrestricted rewrite parser described in Kaplan (1970) -- to derive the deep structure representations of object language strings as relational statements for semantic processing.

Before concluding this discussion on the inter-relations of linguistics and information science, I should mention two other papers concerning the integration of linguistic theory and information science practices which are also relevant here. Mey (1970) suggests an integrating concept in terms of a theory of computational linguistics, stating that attempts at computerization of recognition procedures based on a 'generative semantics' approach (Lakoff 1968; McCawley 1968) could converge with the practical efforts of information scientists toward semantic analysis.

LINGUISTICS AND INFORMATION SCIENCE: SUGGESTIONS FOR A 'METAPRACTICE'

In an earlier paper on linguistics and automated language processing (Montgomery 1969), I noted that the common interest of both linguists and automated language processing specialists in natural language could offset

their divergent analytical approaches to language and emphasized the necessity of mutual cooperation in language processing projects.

To this author, the notion of 'metapractice' -- or interdisciplinary developments including both linguists and information scientists -- is equally as important as the notion of 'metatheory'. Linguists and information scientists have much to learn about natural language and much to learn from each other; the best learning environment could be provided by a joint venture involving a natural language information system for a particular universe of discourse.

For it appears that information science has gone about as far as it can go without linguistics, and conversely. The library of the future can be expected to be very different from the library of today. Many document collections may be replaced by data banks created from natural language text through powerful procedures for content analysis and representation -- e.g., the one outlined in the latter part of this paper. Such procedures involve sophisticated techniques of syntactic and syntactic analysis, and require linguists as well as information scientists for both research and development phases of system construction.

While information scientists have been concentrating on brute force statistical methods or on data management

systems minus a content analysis component (as discussed in the first section), most linguists have been totally ignoring statistics and holding to the notion that one dubious counterexample undermines a theory. After several years of internecine strife in linguistic theory, it seems clear that linguists need to beat their swords into plowshares in the service of some compelling cause.

I suggest that this cause could be a cooperative venture involving linguists and information scientists with the objective of specifying encyclopedic knowledge, based on the metatheoretical assumptions outlined above and within the framework of the natural language information system concept presented in Figures 1, 2, and 3.

The preliminary stage of such a venture would be concerned with the selection of a particular subset of the encyclopedia, the outlines of which are given more or less clearly in terms of a document classification schema. Assuming a machine-readable collection of object language materials and a morphological analyzer of the type described in the beginning of the second section, the initial phase of the development would involve computer analysis of the object language data in order to isolate morphologically defined phrases occurring with significant frequency, as in the 'theoretical linguistics' example in the second section,

as well as statistically significant single words. These data provide a basis for the construction of a natural language thesaurus as described in Montgomery et al 1968 -- i.e., an extension of the document classification schema to include the various syntactically and semantically distinct natural language equivalents (synonyms and hypernyms) of the classificatory terms. Such a device simultaneously provides a detailed specification of the given subset of the encyclopedia for use in elaboration of the metalanguage and the basis for an automated indexing application involving large enough files to avoid the problems resulting from trivially small data bases, noted above in a critique of computational linguistic models.

Thus, the second phase of the project involves research on the one hand and development on the other. One group of linguists and information scientists will be engaged in meticulous research aimed at defining the 'logical semantic' and 'linguistic semantic' relations which are implicitly contained in the natural language thesaurus.

The theoretical foundations of this research are the role notions of Fillmore and the principles of relational logic, as shown in the 'break', 'criticize', and 'compile' examples presented above. In the definition of classes,

an additional notion might also prove useful. This notion (from Zadeh 1970) concerns the definition of classes the boundaries of which are not clearcut, e.g., the class of individuals which can be described by a name such as 'green', 'tall', etc. Membership in one of these vaguely defined classes or 'fuzzy sets' is given by a number in a defined interval representing the 'grade of membership' in the sets, rather than by a binary feature. This approach suggests a method for dealing with logical implication in propositions like Leech's example: '"A tall four-year old child lives next door."' In terms of 'quantitative fuzzy semantics' the presumably low-valued 'grade of membership' of 'four-year old child' in the set 'tall' would block the inference that 'a tall person lives next door', since the 'grade of membership' of 'person' in 'tall' is given by a higher value.

At the same time that one group of linguists and information scientsts is engaged in this intensive research effort toward definition of a metalanguage, a second group of researchers will be engaged in evaluating the analytical data generated by operations of the automated indexing system based on the natural language thesaurus. The object of these investigations is to identify additional natural language realizations of concepts defined in the document

classification system, as well as to determine relative frequencies of occurrence of the various distinct syntactic and semantic realizations of particular classificatory concepts. This type of data will be used in developing a tentative set of analytical priorities for explicating the metalanguage and defining the procedure relating object language and metalanguage.

These data will also provide feedback information to the automated content analysis procedure (as shown in Figures 1 and 2), improving the indexing by adding new terms to the thesaurus and providing approximate values for weighting entries in terms of their information content. At some point in this phase of development, a retrieval capability can be added to the automated system by treating information requests as short documents which are indexed and passed against an invented file to obtain responsive documents.

Based on the progress of the research effort to explicate the metalanguage, successive stages of the development will generate fact files of relational statements derived from input text, and will provide answers to questions requiring inductive and deductive logic. Document abstracts which are coherent, concise paraphrases of document content can also be created, and 'high quality' machine translation becomes possible, assuming a parallel apparatus for

the target language as well as an interface between the source language analyzer and the target language generator.

Based on the survey of theory and practice involving natural language presented in this paper, the construction of the natural language information system outlined in the preceding paragraphs is clearly not a trivial undertaking, for we are attempting to build a device for 'understanding' natural language text before we fully understand natural language. However, the state of the art can only be advanced by attempting achievements which are beyond it.

I therefore suggest that the state of the art in information science, linguistics, and computational linguistics can be substantially advanced by the proposed joint attack upon the problem of understanding natural language in both of the above senses, using what is known to reach the unknown.

FOOTNOTES

[1] The Soviet linguistic tradition of lexicology (see Weinreich 1963) constitutes the only exception to this generalization, as noted below in the section entitled 'Syntax versus Semantics'.

[2] For example, Montgomery and Swanson (1962) document a situation in which a permuted title index would appear to produce results equivalent to the output of a human indexing operation. We also noted the inadequacy of such an index for retrieval.

[3] Thus document subsets -- sentences or paragraphs extracted from a document, or extract-type abstracts -- may contain anaphoric references, while other types of abstracts, facts, and data items are presumably self-contained.

Some of my colleagues may object to a system concept which treats facts (assuming these are single natural language sentences) and documents (ordered sets of natural language sentences) analogously. However -- quite aside from the expository advantage gained by generalizing the explanatory concept - there are some good and cogent reasons for considering fact and document processing systems as variations on a single theme. First, there is the fact that it is not really known how much improvement in retrieval effectiveness might be realized through incorporating in a document retrieval system the more powerful content analysis operations deemed necessary for fact retrieval. For example, though a syntactic analyzer of some sort is generally regarded as necessary in a fact retrieval system, it is thought to be unnecessary in document retrieval, on the basis of a few questionable experiments (see the discussion of the SMART system under Automated Approaches Combining Syntactic and Semantic Analysis). It would seem that in an information system where the content of the document collection is relatively homogeneous, a more powerful content analysis operation -- that is, one including a capability for syntactic analysis-- would greatly enhance precision. In any case, there are some specialized applications which essentially require syntactic

analysis for effective functioning. One such application is an automated indexing system for the American College of Radiology, in which radiological reports are indexed by an anatomical term and a pathological term unless the pathological term is negated. These requirements obviously entail an analysis of the relations between concepts, rather than a simple matching of text words against a list of concepts.

A second reason for treating documents and facts analogously derives from a long-range view of the nature and function of information systems. It appears likely that many document collections will be replaced by data banks or fact files which can provide specific answers to input queries. However, it is extremely unlikely that individual facts will be input one at a time as this would constitute an appallingly inefficient mode of acquisition. Presumably, the source of data for the fact files of the future will be natural language text, which will be operated upon by a powerful content analyzer of the type described in the last three sections of this paper, reducing the ordered sets of natural language sentences to ordered sets of relational statements comprising the fact files. Thus it is reasonable to predict that all natural language information systems of the future will be text processing systems of some sort and the distinction between the fact and document processing systems of today will become obsolete.

[4] A good system of this type which features rapid on-line access and more flexibility of content parameters is ORBIT II (System Development Corporation, 1970).

[5] A few significant exceptions to this general trend are more appropriately discussed in the next section, since they include automated content analysis of some type.

[6] This concern dates from about 1955, although the actual publication date of "Syntactic Structures" was 1957.

[7] These symbols are from Kuno and Oettinger 1963.

[8] This account of transformational grammar is necessarily somewhat simplified.

9
A similar concept was earlier reported in Kuno (1965).

10
Inasmuch as only the first of three projected sections of Noël's dissertation is available to me at present, I am extrapolating from his metatheoretical proposals as presented in that section to some extent, and may thus be guilty of misrepresenting him on some points. His ideas are extremely interesting and are of value to both linguistics and information science.

11
Unlike Noël, Petöfi does not attempt to define a metatheory relating the two disciplines. In terms of Noël's metatheory (see the fourth section of this paper for a discussion of Noël's metatheoretical proposals), Petöfi is rather concerned with specification of the metalanguage and the procedure than with the metatheoretical framework in which they are included.

12
An earlier version of this parser is used in the REL system (Dostert and Thompson 1971; Thompson et al 1969).

13
Resemblances between Schank and Tesler's list of experiences, Simmon's 'Semantic Event Forms', and Quillian's linked 'units' and 'properties' are evident. Quillian's 'Teachable Language Comprehender' also operates with a human 'teacher', who monitors progress in the analysis of text, 'teaching' the system new concepts and supplying syntactic information as necessary.

14
REL and CONVERSE are exceptions to this generalization, as noted in the discussion of these systems.

15
More recent efforts include the work of Apresyan, Zholkovskij, and Mel'chuk (1968, and earlier) and a collection of essays dealing with a variety of problems associated with natural language data processing (Shrejder et al 1967).

16
Some modifications made by Fillmore since publication of the 1968 article are incorporated in this list.

17
It goes without saying that these propositions are also analyzable in terms of relational logic and the logic of classes. Compare Whitehead and Russell (1950, pp. 231ff.).

18
Underlines denote

19
It would appear that this relationship is redundantly specified, if Fillmore's relational statements of role structure are taken as primitive. However, inasmuch as his notions are here extended to include presupposition--as well as other 'logical semantic' relations, and additional extensions may be necessary, it seems more prudent to supply 'labeling' relations, rather than assume that the label follows from the specification of possible role relations. In this sense, labeling relations can be regarded as axioms relating the primitive role concepts of Agent, Instrument, etc., and the predicate concept.

20
See footnote 18.

21
Since the meaning of the verb 'compile' essentially includes the Instrument 'compiler' as well as the Goal 'language', a set of sentences involving some qualification of the meaning of 'compile'--say, negation--would occur more frequently than the cited examples. The cited forms are presented as analogs to the sentences with the verb 'break' (examples (d), (g), (h), and (i) above), and are therefore not qualified by negation or embedding.

22
This example is for illustrative purposes only, and does not represent any formal attempt at specifying a metalanguage for computer science.

23
The deep structure representation of sentences containing the verb 'use' and synonymous sentences where the meaning of 'use' is given by an Instrument relation has been the topic of some debate among linguists (see Lakoff 1968). If separate deep structures are posited for both (as shown below in (i) and (ii)), a statement indicating the semantic equivalence of the two statements is necessary.

(i) A use I $\xrightarrow{\text{Result}}$ I predicate O

Students use compiler $\rightarrow$ Compiler compile programs

(ii) A predicate O I

Students compile programs (with) compiler

On the other hand, if both are represented by the (i) structure (as suggested in Lakoff 1968), some transformations to derive (ii) from (i) will be necessary--i.e., the verb use and its object (the argument of the Instrument relation, as designated above) must be deleted and replaced by the statement representing the result of the activity.

[24] See footnote 10.

[25] For example, see Lyons (1968), Wilks (1968), Werner (1969), Noël (1970).

REFERENCES *

APRESYAN, YU D., ZHOLKOVSKIJ, A. K., MEL'CHUK, I. A. O sisteme semanticheskogo sinteza. III. Obraztsy slovarnykh statej. Nauchno-Tekhnicheskaya Informatsiya, Ser. 2, No. 11, 1968, pp. 8-21.

BAXENDALE, P. B. and CLARKE, D. C. Documentation for an economical program for the limited parsing of English: lexicon, grammar, and flowcharts. IBM San Jose Research Laboratory, San Jose, California, August 16, 1966.

BECKER, J. D. The modeling of simple analogic and inductive processes in a semantic memory system. In Proceedings of the International Joint Conference on Artificial Intelligence, ed. by D. E. Walker and L. Norton, April 1969, pp. 655-668.

BELLERT, I. On the use of linguistic quantifying operators in the logico-semantic structure representation of utterances. International Conference on Computational Linguistics, Preprint No. 28, 1969.

BIERWISCH, M. Some semantic universals of German adjectives. Foundations of Language, Vol. 3, 1967, pp. 1-36.

BOBROW, D. G. and FRASER, B. An augmented state transition network analysis procedure. In Proceedings of the International Joint Conference on Artificial Intelligence, ed. by D. E. Walker and L. Norton. April 1969, pp. 557-567.

BOHNERT, H. G. and BACKER, P. O. Automatic English-to-logic translations in a simplified model. Final report. March 1966. IBM Watson Research Center, Yorktown Heights, N. Y. (AFOSR66-1727) (AD 637227).

BRINER, L. L. and CARNEY, G. J. SYNTRAN/360, a natural language processing system for preparing text references and retrieving text information. IBM Corp., Gaithersburg, Md., 1968. (Preprint)

BÜNTING, D. K. Empirical investigation of German word derivation with the aid of a computer. International Conference on Computational Linguistics, Preprint No. 30, 1969.

CHAFE, W. L. Meaning and the structure of language. Chicago, the University of Chicago Press, 1970.

CHAPIN, PAUL G. and NORTON, LEWIS M. A procedure for morphological analysis. Presented at the Annual Meeting of the Association for Computational Linguistics, Urbana, Illinois, July 1968. Technical report, The MITRE Corporation, Bedford, Mass., July 1968.

CHERNYJ, A. I. Obshchaya metodika postroeniya tezaurusov. Nauchno-Tekhnicheskaya Informatsiya, Ser. 2, No. 5, 1968, pp. 9-33.

CHOMSKY, N. Syntactic structures. The Hague, Mouton and Company, 1957.

CHOMSKY, N. Formal properties of grammars. In Handbook of Mathematical Psychology, ed. by R. D. Luce, R. R. Bush and E. Galanter, New York, John Wiley and Sons, 1963. Vol. II, pp. 323-418.

CHOMSKY, NOAM. Aspects of the theory of syntax. The MIT Press, Cambridge, Mass. 1965.

CLARKE, D. C. and WALL, R. E. An economical program for limited parsing of English. Proceedings of the FJCC, 1965, pp. 307-316.

COLES, L. S. Talking with a robot in English. In Proceedings of the International Joint Conference on Artificial Intelligence, ed. by D. E. Walker and L. Norton, April 1969, pp. 587-596.

CROS, R. C., GARDIN, J. C. and LEVY, F. L'automatisation des recherches documentaires: un modèle général, le SYNTOL. Paris, Gauthier-Villars, 1964.

CULICOVER, P., KIMBAL, J., LEWIS, C., LOVEMAN, D. and MOYNE, J. An automated recognition grammar for English. Technical Report FSC 69-5007, IBM Corporation, Cambridge, Massachusetts, July 1969.

DOLBY, J. L. and RESNIKOFF, H. L. The English word speculum. Mouton. The Hague. 1967, Vol. I, II, III, IV, V.

DOSTERT, B., and THOMPSON, F. B. English for the computer: The interface of syntax and semantics. (For presentation at The International Federation for Information Processing Congress at Ljubljana, Yugoslavia, August 23-28, 1971.)

EARL, L. L. Automatic determination of parts of speech of English words. Mechanical Translation and Computational Linguistics, Vol. 10, Nos. 3 and 4, September and December 1967, pp. 53-67.

EARLEY, J. An efficient context-free parsing algorithm. Communications of the ACM, Vol. 13, No. 2, February 1970.

FILLMORE, CHARLES J. The Case for Case. In: Bach, Emmon and Robert T. Harms, Eds. Universals in linguistic theory. Holt, Rinehart and Winston, Inc., New York. 1968.

FILLMORE, C. J. Verbs of judging: an exercise in semantic description. Papers in Linguistics (Florida State University), Vol. 1, No. 1, July 1969, pp. 91-117.

GARVIN, P. L., BREWER, J. and MATHIOT, M. Predication typing: a pilot study in semantic analysis. Language, Vol. 43, No. 2: Language Monograph No. 27, Part 2, June 1967.

GRUBER, J. Functions of the lexicon in formal descriptive grammars. (TM-3770). System Development Corp., Santa Monica, Calif., December 1967.

GREEN, C. C., and RAPHAEL, B. The use of theorem-proving techniques in question-answering systems. In: Proceedings of The National Conference of the Association for Computing Machinery, August 1968, pp. 169-181.

HARRIS, Z. Mathematical structures of language. Wiley, New York, 1963.

HAYS, D. G. Dependency Theory: a formalism and some observations. Language, Vol. 40, No. 4 (October-December 1964), pp. 511-525.

JONES, K. SPARCK and NEEDHAM, R. M. Automatic term classification and retrieval. Information Storage and Retrieval 4, 2 (June 1968), pp. 91-100.

JONES, PAUL E., CURTICE, ROBERT M., GIULIANO, VINCENT E. and SHERRY, MURRAY E. Application of statistical association techniques for the NASA document collection. Arthur E. Little, Inc., Cambridge, Mass., February 1968. NASA CR-1020. Prepared under Contract No. NASw-1051.

JUILLAND, A. G. Dictionnaire inverse de la langue française. The Hague, Mouton, 1965.

KAPLAN, R. M. The MIND system: a grammar-rule language. The RAND Corporation, Santa Monica, California, RM-6265/1-PR, April 1970.

KARLGREN, H. The possibility and/or necessity of CS-rules in categorial grammar. KVAL (Research Group for Quantitative Linguistics), Stockholm, Interim Report No. 6, 1968.

KARTTUNEN, L. Discourse referents. International Conference on Computational Linguistics, Preprint No. 70, 1969.

KATZ, J. J. The philosophy of language. New York, Harper and Row, 1966.

KAY, M. and SU, S. Y. W. The MIND system: the structure of the semantic file. The RAND Corporation, Santa Monica, California, RM-6265/3-PR, 1970.

KELLOGG, CHARLES H. A natural language compiler for on-line data management. System Development Corp., Santa Monica, Calif., 30 August 1968, 51 p. (SP-3160). For presentation at the 1968 Fall Joint Computer Conference, San Francisco, Calif., 9-11 December 1968.

KIEFER, F. Mathematical linguistics in Eastern Europe. New York, American Elsevier, 1968.

KLINGBIEL, P. H. Machine-aided indexing. Defense Documentation Center Technical Progress Report DDC-TR-69-1, June 1969.

KRAVCHENKO, N. D. Opisanie algoritma avtomaticheskogo vydeleniya iz teksta sushchestvitel'nykh i prilagatel'nykh i formirovaniya imen. Nauchno-Teknicheskaya Informatsiya 1968, Series 2, No. 2, pp. 13-18.

KUHNS, J. L. Logical aspects of question-answering by computer. (Presented at the Third International Symposium on Computer and Information Sciences, Miami Beach, Florida, December 1969.)

KUNO, S. and OETTINGER, A. G. Multiple-path syntactic analyzer. Information Processing 1962 (Proceedings of the IFIP Congress 1962). Ed. Cicely M. Popplewell. 1963, Amsterdam, North-Holland Publishing Company.

LAFFAL, J. Total or selected content analysis. International Conference on Computational Linguistics, Preprint No. 24, 1969.

LAKOFF, G. Instrumental adverbs and the concept of deep structure. Foundations of Language, Vol. 4, No. 1, pp. 4-29, January 1968.

LEECH, G. N. Towards a semantic description of English. Bloomington, Indiana University Press, 1970.

LEVIEN, R. E. and MARON, M. E. A computer system for inference execution and data retrieval. Communications of the ACM, 10:11 (November 1967) pp. 715-721.

LYONS, JOHN. Introduction to theoretical linguistics. Cambridge University Press. 1968.

MARTINS, G. R. The MIND system: The morphological analysis program. The RAND Corporation, Santa Monica, California, RM-6265/2-PR, April 1970.

MCCAWLEY, JAMES D. the role of semantics in a grammar. In: Bach, Emmon and Robert T. Harms, Eds. Universals in linguistic theory. Holt, Rinehart and Winston, Inc. New York. 1968.

MEY, J. Towards a theory of computational linguistics. (Presented at the annual meeting of the Association for Computational Linguistics, The Ohio State University, July 1970).

MONTGOMERY, CHRISTINE A. "Automated Language Processing" in Annual Review of Information Science and Technology, Carlos A. Cuadra, Ed., Vol. 4, Encyclopedia Britannica Press, 1969, pp. 145-174.

MONTGOMERY, C. A. Linguistics and automated language processing. International Conference on Computational Linguistics, Preprint No. 41, 1969.

MONTGOMERY, C. A. and SWANSON, D. R. Machinelike indexing by people. American Documentation, Vol. 13, No. 4, October 1962.

MONTGOMERY, C. A., WORTHY, R. M., REITZ, G. et al. Optical Character Reader Applications Study. August 1968. The Bunker-Ramo Corp., Canoga Park, California. Final technical report prepared for Rome Air Development Center, Griffiss Air Force Base, N. Y., under Contract F30602-67-C-0169. Covers period Jan. 24, 1967-Aug. 24, 1968.

NOËL, J. A. A semantic analysis of abstracts around an experiment in mechanized indexing. Part I, draft of doctoral dissertation.

NOËL, J. L'indexation mecanise ede resumes anglais: quelques hypotheses et analyses semantiques. 44p, mimeographed. October 1966.

OTRADINSKIJ, V. V. and KRAVCHENKO, N. D. Ob odnom metode indeksirovaniya pol'nykh tekstov. Nauchno-teknicheskaya Informatsiya 1969, Series 2, No. 7, pp. 16-22.

SCHANK, ROGER C. and TESLER, LAWRENCE G. A conceptual parser for natural language. Report: Computer Science Department. School of Humanities and Sciences, Stanford University, January 1969.

SCHWARCZ, R. M. Towards a computational formalization of natural language semantics. International Conference on Computational Linguistics, Preprint No. 29, 1969.

SHAPIRO, S. C. and WOODMANSEE, G. H. A net structure based relational question answerer: description and examples. In Proceedings of the International Joint Conference on Artificial Intelligence, ed. by D. E. Walker and L. Norton, April, 1969, pp. 325-346.

SHREJDER, YU. A., PROBST, M. A., BORSHCHEV, V. B., and EFIMOVA, E. N. Semioticheskie problemy avtomatizirovannoj obrabotki informatsii. Akademiya Nauk SSSR, Moscow, 1967.

SIMMONS, R. F. Natural language question-answering systems: 1969. Communications of the ACM, Vol. 13, No. 1, 1970, pp 15-30.

SIMMONS, ROBERT F., BURGER, JOHN F., SCHWARCZ, ROBERT M. A. computational model of verbal understanding. System Development Corp., Santa Monica, Calif. 30 April 1968, 52 p. (SP-3132).

SU, S. Y. W. and HARPER, K. E. A directed random paragraph generator. The RAND Corporation, Santa Monica, California, RM-6053-PR, July 1969.

SYSTEM DEVELOPMENT CORPORATION. ORBIT II system information. SDC Document SP 3563 (System Development Corporation, Santa Monica, California), November 1970.

TESNIERE, L. Elements de syntaxe structurale. Paris. 1959.

THARP, A. L. and KRULEE, G. K. Using relational operators to structure long-term memory. In Proceedings of the International Joint Conference on Artificial Intelligence, ed. by D. E. Walker and L. Norton, April 1969, pp. 579-586.

THOMPSON, F. G., LOCKEMANN, D. C., DOSTERT, B., and DEVERILL, R. S. REL: a rapidly extensible language system. In: Proceedings of the National Conference of the Association for Computing Machinery. August 1969, pp. 399-417.

THORNE, J., BRATLEY, P. and DEWAR, H. The syntactic analysis of English by machine. In Machine Intelligence 3, ed. by D. Michie. American Elsevier Press, New York, 1968.

VASILIU, E. The 'time category' in natural languages and its semantic interpretation. International Conference on Computational Linguistics, Preprint No. 60, 1969.

VARGA, D. Postroenie novoj analiziruyushchej sistemy predlozhenij. Nauchno-Tekhnicheskaya Informatsiya, series 2, no. 4 (1968), pp. 17-23.

VAUQUOIS, B., VEILLON G., NEDOBEJKINE, N. and BOURGUIGNON, C. Une notation des textes hors des contraintes morphologigues et syntaxiques de l'expression. International Conference on Computational Linguistics, Preprint No. 17, 1969.

VON GLASERSFELD, E. Semantics and the syntactic classification of words. International Conference on Computational Linguistics, Preprint No. 22, 1969.

WEINREICH, U. Lexicology. In Current Trends in Linguistics, Vol. 1. Soviet and East European Linguistics, The Hague, Mouton and Company, 1963, pp. 60-93.

WERNER, O. On the universality of lexical/semantic relations. (Working paper presented at the American Anthropological Association Annual Meeting, November 1969).

WEST, L. E. SPIRAL (Sandia's Program for Information Retrieval and Listing). Sandia Laboratories Research Report No. SC-RR-68-819-C, December 1968.

WHITEHEAD, A. N., and B. Russell, Principia Mathematica (Second Edition), Vol. I, Cambridge University Press, 1950

WILKS, YORICK. Computable semantic derivations. System Development Corporation, Santa Monica, California, 15 January 1968, 160 p. (SP-3017).

WILLIAMS, T. M. Case variables and case description in a reticular logistic grammar. (Working paper). Graduate Library School, The University of Chicago, Chicago, Illinois, September 1966.

WOODS, W. A. Procedural semantics for a question-answering machine. In: AFIPS Conference Proceedings, Vol. 33, Part 1, 1968 Fall Joint Computer Conference, pp. 457-471.

WOODS, W. A. Augmented transition networks for natural language analysis. Report No. CS-1 to the National Science Foundation, Aiken Computation Laboratory, Harvard University, January 1970.

WORTH, D., KOZAK, A. and JOHNSON, D. Russian derivational dictionary. The RAND Corporation, Santa Monica, California, R-489-PR. January 1970.

ZADEH, L. A. Quantitative fuzzy semantics. Electronics Research Laboratory, University of California, Berkeley, Memorandum No. ERL-M281, August 1970.

ZWICKY, A. M., FRIEDMAN, J., HALL, B. C., and WALKER, D. E. The MITRE Syntactic Analysis Procedure for Transformational Grammars. Proceedings of The Fall Joint Computer Conference, Washington, D. C. Spartan Books, 1965, pp. 317-326.

* References identified as preprints of the 1969 International Conference on Computational Linguistics are available in limited quantities from Dr. Hans Karlgren, KVAL (Research Group for Quantitative Linguistics), Sodermalmstorg 8, Stockholm, Sweden.